
This is the **published version** of the article:

Martín Rivas, Francisco; Ferrero Beato, Ignacio. Desarrollo de una tecnología Smartphone para el estudio del uso y la percepción del espacio público. 2019. 75 p.

This version is available at <https://ddd.uab.cat/record/214372>

under the terms of the  license



Desarrollo de una tecnología Smartphone para el estudio del uso y la percepción del espacio público

*Máster en Geoinformación, Desarrollo de
geoaplicaciones*

Proyecto de Final de Máster

Universitat Autònoma de Barcelona

Francisco Martín Rivas



Histórico del documento

Versión	Fecha	Autor	Acción
1.00	10/04/2019	Fran Martín	Redacción introducción y descripción
1.01	17/04/2019	Fran Martín	Redacción descripción general
1.02	23/04/2019	Fran Martín	Elaboración diagramas casos de uso
1.03	24/04/2019	Fran Martín	Elaboración especificaciones casos de uso
1.04	25/04/2019	Fran Martín	Elaboración requisitos
1.05	26/04/2019	Fran Martín	Implementación de la metodología de Trabajo
1.06	27/04/2019	Fran Martín	Análisis de alcance y análisis de riesgo
1.07	07/06/2019	Fran Martín	Reestructuración del documento
1.08	11/06/2019	Fran Martín	Corrección requisitos; corrección introducción; introducción diagramas casos de uso
1.09	12/06/2019	Fran Martín	Finalización introducción; corrección diagramas
1.1	13/06/2019	Fran Martín	casos de uso; corrección características. Redacción Implementación
1.11	14/06/2019	Fran Martín	Introducción esquema base de datos, tecnologías empleadas
1.12	15/06/2019	Fran Martín	Finalización tecnologías empleadas, portada y bibliografía
1.13	16/06/2019	Fran Martín	Finalización interfases de usuario
1.14	21/06/2019	Fran Martín	Finalización primer borrador
1.15	26/06/2019	Fran Martín	Redacción conclusiones y elaboración glosario de elementos

Índice

RESUMEN	6
1 INTRODUCCIÓN	8
1.1 Propósito del documento	8
1.2 Contexto	8
1.3 Marco institucional	9
1.4 Abasto	9
1.5 Glosario de términos	9
2 OBJETIVOS	10
3 DESCRIPCIÓN GENERAL	10
3.1 Perspectiva del producto	10
3.2 Arquitectura del sistema	10
3.3 Funciones del sistema	11
3.4 Casos de uso del sistema	11
3.5 Tipos de usuario y características	11
3.6 Entorno operativo	12
3.7 Restricciones de diseño e implementación	12
3.8 Documentación de Usuario	13
3.9 Asunciones y dependencias	13
4 METODOLOGÍA DE TRABAJO	13
4.1 Propósito del apartado	13
4.2 Contexto de la metodología	14
4.3 Alcance	14
4.4 Metodología	14
4.4.1 Fundamentos	14
4.4.2 Responsabilidades	15
4.4.3 Planificación	15
4.4.4 Pila de producto	16
5 ANÁLISIS DE REQUISITOS	17
5.1 Propósito del apartado	17
5.2 Alcance	17
5.3 Características del sistema	17
5.4 Diagrama de flujo de procesos del sistema	18
5.5 Casos de uso del sistema	19
5.5.1 Definición de los casos de uso del sistema	20
5.6 Requisitos funcionales	25
5.6.1 Requisitos de información	25

5.6.2 Requisitos de conducta	25
<i>5.7 Requisitos no funcionales</i>	31
5.7.1 Requisitos de fiabilidad	31
5.7.2 Requisitos de usabilidad	31
5.7.3 Requisitos de mantenibilidad	31
5.7.4 Requisitos de portabilidad	32
5.7.5 Requisitos de seguridad	32
5.7.6 Requisitos técnicos	32
5.7.7 Requisitos de integración	33
6 ANÁLISIS DEL SISTEMA	35
<i>6.1 Propósito del apartado</i>	35
<i>6.2 Análisis de alcance</i>	35
<i>6.3 Análisis de riesgos</i>	36
7 DISEÑO	38
<i>7.1 Propósito del apartado</i>	38
<i>7.2 Prototipado de las interfaces de usuario</i>	38
<i>7.3 Base de datos del sistema</i>	38
7.3.1 Definición de la base de datos	38
7.3.2 Diseño de la base de datos	39
8 IMPLEMENTACIÓN	40
<i>8.1 Propósito del apartado</i>	40
<i>8.2 Tecnologías empleadas</i>	40
8.2.1 Desarrollo web	40
8.2.2 Frameworks y bibliotecas	42
8.2.3 Entorno y plataforma de desarrollo	42
<i>8.3 Estructura de directorios</i>	42
8.3.1 Cliente móvil	42
8.3.2 Cliente de escritorio	44
<i>8.4 Implementación de la base de datos</i>	44
<i>8.5 Especificaciones empleadas</i>	44
8.5.1 Formato de almacenamiento de datos	44
8.5.2 Análisis geoespacial	46
<i>8.6 Precisión de posicionamiento</i>	47
9 RESULTADOS	47
<i>9.1 Vistas de usuario</i>	47
9.1.1 Cliente móvil	47
9.1.2 Cliente de escritorio	51
<i>9.2 Obtención de datos</i>	51
<i>9.3 Retorno de datos</i>	52
<i>9.4 Precisión</i>	<i>¡Error! Marcador no definido.</i>
10 CONCLUSIONES	55

11 BIBLIOGRAFÍA	56
11.1 Libros y artículos	56
11.2 Sitios web	56
11.3 Aplicaciones móviles de referencia	57
ANEXOS	58
GLOSARIO DE ÍNDICES	62
Tablas	63
Figuras	66

RESUMEN

Un EMA, o *Environmental Moment Assesment*, es un método de obtención de información del estado físico o mental de una persona que, con la irrupción de los teléfonos móviles, ha visto la oportunidad de integrar ambos conceptos con el objetivo de desarrollar herramientas de recogida de datos en tiempo real que sean agradables y fáciles de utilizar para los usuarios. En este contexto surge la aplicación para teléfonos móviles *Semper ambulans* como una herramienta útil de obtención de información, en términos de cómo los habitantes de la ciudad de Barcelona utilizan el espacio público y la percepción que tienen del mismo. De esta manera, y si bien dicha aplicación podrá ser descargada y utilizada por cualquier persona, la explotación de los datos recogidos se en Mobilitat, Transport i Territori, del Departament de Geografia de la Universitat Autònoma de Barcelona, el cual es el cliente final del presente proyecto. Así, se ha desarrollado un cliente móvil capaz tanto de registrar y georreferenciar las rutas que recorren los usuarios como de mostrar al usuario un formulario sobre cómo le hace sentir el espacio en el que se encuentra, y donde dicho formulario aparece ya sea porque el usuario se ha detenido de manera prolongada o porque está en una zona de interés, como podría ser una plaza, una playa o un parque. A efectos de soporte y monitorización en tiempo real, el grupo investigador del GEMOTT contará también con un cliente de escritorio en el que se puede visualizar e incluso descargar toda la información registrada en la base de datos, en un formato que permite su explotación por clientes SIG como, por ejemplo, QGIS. En definitiva, se ha llevado a cabo tanto el diseño como la implementación completa del sistema, por un lado en relación con la base de datos y por el otro lado con el desarrollo del diseño gráfico y especialmente funcional de los clientes del sistema, obteniendo una herramienta sólida en cuanto a recogida de información que se espera sea utilizada tanto en las líneas de investigación actuales del GEMOTT como en las que puedan iniciarse en el futuro.

Un EMA, o *Environmental Moment Assesment*, és un mètode d'obtenció d'informació de l'estat físic o mental d'una persona que, amb la irrupció dels telèfons mòbils, ha vist l'oportunitat d'integrar tots dos conceptes amb l'objectiu de desenvolupar eines de recollida de dades en temps real que siguin agradables i fàcils d'utilitzar per als usuaris. En aquest context sorgeix l'aplicació per a telèfons mòbils *Semper ambulans* com una eina útil d'obtenció d'informació, en termes de com els habitants de la ciutat de Barcelona utilitzen l'espai públic i la percepció que tenen d'aquest. D'aquesta manera, i si bé dita aplicació podrà ser descarregada i utilitzada per qualsevol persona, l'explotació de les dades recollides amb ella és matèria exclusiva del Grup

Autònoma de Barcelona, el qual és el client final del present projecte. Així, s'ha desenvolupat un client mòbil capaç tant de registrar i georreferenciar les rutes que recorren els usuaris com de mostrar a l'usuari un formulari sobre com li fa sentir l'espai en el qual es troba, i on dit formulari apareix ja sigui perquè l'usuari s'ha detingut de manera perllongada o perquè està en una zona d'interès, com podria ser una plaça, una platja o un parc. A l'efecte de suport i monitoratge en temps real, el grup investigador del GEMOTT comptarà també amb un client d'escriptori en el qual es pot visualitzar i fins i tot descarregar tota la informació registrada en la base de dades, en un format que permet la seva explotació per clients SIG com, per exemple, QGIS. En definitiva, s'ha dut a terme tant el disseny com la implementació completa del sistema, d'una banda en relació amb la base de dades i per l'altre costat en relació amb el desenvolupament del disseny gràfic i especialment funcional dels clients del sistema, obtenint una eina sòlida quant a recollida d'informació que s'espera sigui utilitzada tant en les línies de recerca actuals del GEMOTT com en les quals puguin iniciar-se en el futur.

An EMA, or *Environmental Moment Assessment*, is a method of obtaining information about the physical or mental state of a person, which with the advent of mobile phones, has seen the opportunity to integrate both concepts in order to develop tools for collecting Real-time data that must be user-friendly. In this context, the mobile phone application *Semper ambulans* emerges as a useful tool for obtaining information in terms of how the inhabitants of the city of Barcelona use public space and the feelings they have of it. In this way, and although this application may be downloaded and used by any person, the use of the collected data is the exclusive subject of the Grup d'Estudis en Mobilitat, Transport i Territori, of the Geography Department at the Autonomous University of Barcelona, which is the final client of the present project. Thus, a mobile client has been developed that is capable of both registering and georeferencing the routes that users travel and of showing the user a form about how the space where it is located makes it feel. That form appears either because the user has stopped or because it is in an area of interest, such as a square, a beach or a park. For the purposes of support and monitoring in real time, the GEMOTT research group will also have a desktop client in which they can view and even download all the information registered in the database, in a format that allows its analysis by GIS software as, for example, QGIS. In short, both the design and the complete implementation of the system have been carried out, on the one hand in relation to the database and on the other side with the development of the graphic and especially functional design of the system's clients, obtaining a solid tool in terms of information collection that is expected to be used both in the current research lines of GEMOTT and in those that may be initiated in the future.

1 INTRODUCCIÓN

1.1 Propósito del documento

El presente documento describe el trabajo realizado en el Trabajo de Final de Máster (TFM) del máster en Geoinformación de la Universidad Autónoma de Barcelona (UAB), en su especialización en Desarrollo de Geoaplicaciones. Dicho proyecto consiste en desarrollar una aplicación móvil que permita estudiar el uso y la percepción del espacio público por parte de los habitantes de la ciudad de Barcelona.

1.2 Contexto

El proyecto realizado nace inspirado de un también Trabajo de Final de Máster de características muy similares (Barot, 2018), desarrollado en el curso anterior del mismo máster en Geoinformación y también realizado para el *Grup d'Estudis de Mobilitat, Transport i Territori* de la Universidad Autónoma de Barcelona - ver siguiente apartado - bajo un contexto de investigación científica en el ámbito de la movilidad cotidiana.

Así, el citado proyecto se enmarcó en una línea de investigación relativa al estudio de los entornos saludables para la gente mayor, cuyo objetivo general era explorar cómo las personas percibían los entornos urbanos en sus desplazamientos a pie y qué sensaciones les transmitían los elementos o espacios urbanos. Con este propósito se desarrollaron dos aplicaciones, una a modo de cuestionario para los usuarios llamada *POEfeelings*, y otra a modo de herramienta de control para el equipo del GEMOTT, llamado *POEfeelingsMap*. Dichas aplicaciones permitieron establecer una base teórica y metodológica como punto de partida para nuevos trabajos, dando pie a un nuevo proyecto como es el expuesto. Aun así, es cierto que no se puede entender como una ampliación dado que no se han mantenido exactamente las mismas áreas temáticas respecto al proyecto inicial.

En este contexto, y visto el estado de la cuestión, nace el proyecto de *Desarrollo de una tecnología Smartphone para estudiar el uso y percepción del espacio público* como Trabajo de Final de Máster, en un intento de asentar las bases establecidas y explorar de manera real la posibilidad de implantar un sistema como el que se propone. Así, la aplicación se idea como una herramienta útil y de explotación realista cuya función principal es la obtención de datos fiables y coherentes relativos al uso y percepción del espacio público por parte de los habitantes de la ciudad de Barcelona, radicando estos datos básicamente en registros de rutas y registros de formularios. De esta manera, se espera que esta herramienta de recopilación de información permita obtener datos sólidos que sustenten no solamente el eje de investigación en el cual se engloba la aplicación, como es el estudio del uso y percepción del espacio público, sino también las diferentes líneas de investigación que el equipo investigador del GEMOTT considere oportunas.

Más aún, y para entrar más en profundidad en la descripción del sistema desarrollado, el mismo se engloba dentro del concepto de *Enviromental Momentary Assesment*, o simplemente EMA. Un EMA, también conocido como *in the moment assesment* o *experience sampling*, es un método ampliamente utilizado en campos como la dinámica territorial o incluso la psicofarmacología clínica (Moskowitz & Young, 2006) que permite obtener información sobre el estado físico y mental de una persona en el momento actual, en contra del método tradicional de la evaluación retrospectiva (Marquet *et al.*, 2017; Van de Ven *et al.*, 2017). De esta manera, la irrupción de los *smart phones* ha permitido el surgimiento de una tecnología muy válida en cuanto a la obtención de datos de tipo EMA se refiere, puesto que se los teléfonos móviles se han convertido en la interfaz dominante entre los usuarios y la infraestructura tecnológica que nos rodea. Este hecho ha motivado el desarrollo de este sistema como un método de obtención de información continua tanto en la escala temporal como en la escala espacial.

1.3 Marco institucional

El proyecto ha sido realizado para el *Grup d'Estudis de Mobilitat, Transport i Territori*, a partir de ahora denominado simplemente como *GEMOTT*. El GEMOTT es un equipo interdisciplinario de investigadores, ubicado en el Departamento de Geografía de la Universidad Autónoma de Barcelona, cuyo eje básico de investigación se centra en la movilidad cotidiana de las personas desde una perspectiva territorial, social y medioambiental. Así, el presente proyecto – y en consecuencia, el desarrollo de las aplicaciones descritas – se ubica dentro de uno de los ejes básicos de investigación del grupo, el cual se centra en el estudio del uso y percepción del espacio y el tiempo en las dinámicas de movilidad cotidiana.

Así mismo, y como se ha indicado en el apartado 1.1 (Propósito del documento), el presente proyecto consiste en el trabajo realizado en el Trabajo de Final de Máster del autor del documento, como resultado del módulo integrado por las Prácticas externas obligatorias y el señalado TFM. Por su lado, las Prácticas externas tienen un papel fundamental en el máster, tanto por conformar una experiencia formativa y un primer contacto con el mundo profesional como por servir de base para desarrollar el Trabajo de Final de Máster. Este, a su vez, tiene por finalidad la realización de un trabajo práctico de aplicación real, en este caso de la especialidad en Desarrollo de Geoaplicaciones, con el objetivo de que el alumno demuestre que ha adquirido y sabe integrar las competencias que son objeto del Máster.

1.4 Abasto

El presente documento pretende hacer una descripción precisa y comprensible de la totalidad del proceso que se ha llevado a cabo. En este contexto, se pretende aportar una descripción y visión general del producto – incluyendo, entre otros, los objetivos que espera satisfacer el proyecto, sus características y el contexto de desarrollo del mismo -, especificar y analizar los requisitos del sistema – tanto funcionales como no-funcionales -, precisar de manera abstracta pero detallada los componentes - software - que forman el producto, definir el funcionamiento y los principales casos de uso de la aplicación, describir de manera detallada el proceso de implementación del producto y, finalmente, mostrar los resultados obtenidos.

1.5 Glosario de términos

- **SGBD** (*Sistema de Gestión de Bases de Datos*): conjunto de programas que permiten el almacenamiento, modificación y extracción de la información en una base de datos.
- **DOM** (*Document Object Model*): conjunto de utilidades específicamente diseñadas para manipular documentos HTML de manera rápida y eficiente.
- **AJAX** (*Asynchronous JavaScript And XML*): técnica que permite desarrollar aplicaciones web capaces actualizarse continuamente sin tener que volver a cargar la página completa, creando aplicaciones más rápidas y con mejor respuesta a las acciones del usuario.
- **MoSCoW**: técnica de priorización que invita a separar los requisitos o características en cuatro categorías:
 - **M - Must**: Requisitos que debe tener la solución. Son los requisitos mínimos que hacen usable la solución.
 - **S – Should**: Requisitos que debería incluir la solución. Requisitos importantes pero no obligatoriamente necesarios.
 - **C – Could**: Requisitos que podría incluir la solución.
 - **W – Won't**: Requisitos que no se van a hacer pero que en un futuro podrían incorporarse.
- **Parsear JSON**: método para analizar una cadena de texto y transformarla al formato *json*.

2 OBJETIVOS

El objetivo principal del proyecto consiste en desarrollar una aplicación móvil que permita explorar con qué intensidad y por qué motivos las personas utilizan el espacio público. Dicha aplicación debe permitir registrar el recorrido cotidiano a pie de los usuarios, a la vez de preguntar en cada punto de parada o al llegar a determinados sitios una serie de cuestiones referentes al espacio en el que se encuentra el usuario. Así, la aplicación enviará los recorridos geolocalizados y las respuestas de los usuarios a un servidor para que sean almacenados en una base de datos.

Más aún, se debe desarrollar un visor web con el objetivo de dotar a los clientes y administradores de una aplicación de *back-office* que permita la visualización del total de rutas y formularios registrados en la base de datos, en pos de permitir tanto la monitorización en tiempo real de la recepción de las mismas como de constituir una herramienta útil de explotación de estos últimos.

3 DESCRIPCIÓN GENERAL

3.1 Perspectiva del producto

La perspectiva del producto es permitir el estudio de la intensidad y los motivos por los que las personas utilizan el espacio público, obteniendo información coherente y que permita desarrollar conclusiones sólidas y fiables. Se prevé que la aplicación se mantenga operativa y funcional de forma perlongada, de manera que pueda ser utilizada para las líneas y proyectos de investigación que el equipo investigador del GEMOTT considere necesarias como una herramienta útil de obtención de datos.

3.2 Arquitectura del sistema

El sistema por desarrollar consiste en un sistema de arquitectura orientada a servicios, esto es, de tres niveles: cliente o software que representan entidades que necesitan servicios, servidor de aplicación o programas que proporcionan servicios y servidor de datos. En dicha arquitectura, cliente, servidor y base de datos son objetos separados desde un punto de vista lógico que se comunican a través de una red de comunicaciones, con el objetivo de realizar una o varias tareas de forma conjunta. La figura 1 es un esquema resumen de la estructura del sistema.

Así, y en lo relativo a la información, la misma estará compuesta esencialmente por registros de rutas y respuestas a formularios, almacenados en una base de datos de tipo alfanumérico que será gestionada en un entorno único mediante PostgreSQL. Más aún, y mientras que dicha información será visualizable en dos plataformas diferentes, como son un cliente web de escritorio y un cliente web móvil, únicamente será explotable a través de la plataforma de escritorio, mientras que solamente se adquirirá y recogerá información a través de la plataforma móvil.

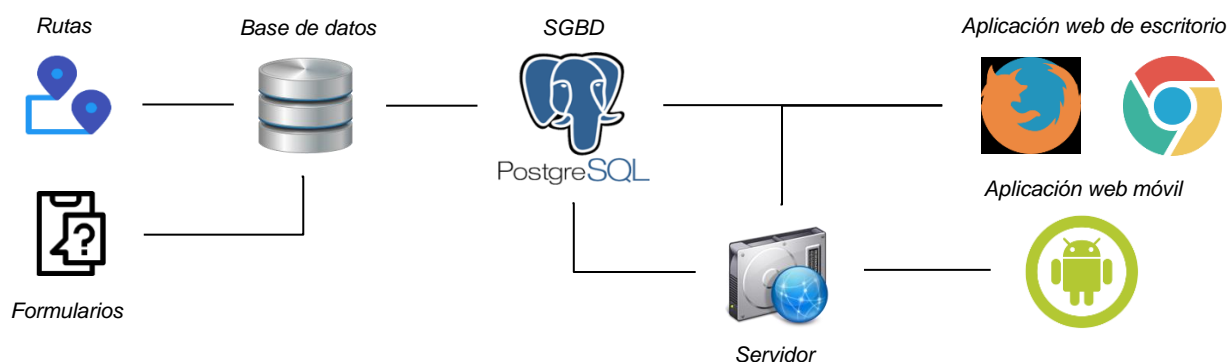


Figura 1. Esquema de la arquitectura del sistema desarrollado. Fuente: elaboración propia.

3.3 Funciones del sistema

Las funcionalidades principales del sistema son todas relativas a la geolocalización de rutas y resolución de formularios por parte de los usuarios de la aplicación móvil, juntamente a la explotación de la totalidad de esta información. De manera concisa, el conjunto de funciones del sistema son las siguientes:

- a) Aplicación móvil
 - I. Actividades principales
 - i. Geolocalizar ruta a pie
 - ii. Mostrar formulario
 - iii. Responder formulario
 - II. Registro de datos
 - i. Enviar ruta geolocalizada al servidor
 - ii. Enviar respuestas del formulario al servidor
 - III. Visualización de datos
 - i. Visualizar rutas recorridas en un visor dinámico
 - ii. Visualizar posición de los formularios resueltos
 - iii. Visualizar respuestas a formularios resueltos
- b) Aplicación web
 - I. Visualización de datos
 - i. Visualizar rutas geolocalizadas registradas en la base de datos
 - ii. Visualizar respuestas a formularios resueltos registradas en la base de datos
 - II. Explotación de datos
 - i. Descarga de datos almacenados en la base de datos

3.4 Casos de uso del sistema

Los casos de uso del sistema se describen en el capítulo 5, *Análisis de requisitos*, en el apartado 5.5, *Casos de uso del sistema*.

3.5 Tipos de usuario y características

Al tratarse de una aplicación sin ánimo de lucro con fines exclusivamente centrados en la investigación, el abanico de usuarios queda relativamente reducido. Considerando esto, se pueden distinguir tres grupos claros de usuario, tal y como queda recogido en la tabla 1.

Grupo	Descripción	Funciones de uso
Desarrolladores	Los miembros del equipo de desarrollo son los encargados de desarrollar las aplicaciones siguiendo un correcto proceso de ingeniería del software. De entre las distintas fases que contempla el proyecto, la prueba del sistema constituye una parte crítica en pos de asegurar el correcto funcionamiento de este. Así, los desarrolladores también se convierten en usuarios, puesto que están obligados a hacer las pertinentes pruebas de funcionamiento.	Desarrollo y <i>test</i> del software.

Usuarios	Los usuarios o <i>peatones</i> son el público objetivo a los cuales está destinada la aplicación móvil. Deben ser capaces de hacer un uso simple de la misma, pudiendo tanto geolocalizar sus rutas a pie como resolver formularios. Disponen, además, de la posibilidad de visualizar las rutas que han recorrido a través un mapa dinámico.	Público objetivo de la aplicación móvil. Interacción con la lógica del software, en concreto con la geolocalización de sus rutas a pie y de la resolución de formularios.
Investigadores	Los investigadores o <i>administradores</i> son los encargados de gestionar y explotar toda la información y datos recogidos a través de la aplicación móvil, en relación tanto a las rutas como a las respuestas de los formularios. Son, además de los clientes finales y público objetivo de la totalidad del sistema, los usuarios de la aplicación web de <i>back-office</i> . Por ello, se les puede considerar el grupo de usuarios principal. Más aún, los investigadores también se pueden convertir en usuarios en el caso que deseen participar en las pruebas de funcionamiento.	Público objetivo de la aplicación web y cliente de la totalidad del sistema. Gestión y explotación de la base de datos. Interacción con la lógica del software, en concreto con la visualización de rutas geolocalizadas y las respuestas a los formularios.

Tabla 1. Definición de los tipos de usuarios del sistema desarrollado, juntamente a sus funciones de uso. Fuente: elaboración propia.

3.6 Entorno operativo

En el caso de la aplicación móvil híbrida, la misma está diseñada para poder ser adaptada a múltiples plataformas móviles sin la necesidad de realizar grandes cambios en el código fuente, aunque sí ajustándolo a las correspondientes diferencias operacionales para cada plataforma. De esta manera, se espera que la aplicación se implemente en Android, con la posibilidad de expandirse al sistema operativo iOS en el futuro.

Por lo que respecta al visor web, al tratarse de aplicación de navegador no existe ningún tipo de restricción respecto a su entorno operativo.

3.7 Restricciones de diseño e implementación

- **Conexión a internet:** la principal restricción a la que se enfrenta la aplicación tiene relación con la conexión a Internet por parte del dispositivo móvil del usuario. Si bien dicha conexión no es estrictamente necesaria para la geolocalización del individuo, sí que es imprescindible tanto para permitir la visualización del mapa base de la aplicación como, aún más importante, para permitir el envío de las rutas geolocalizadas y las respuestas de las preguntas al servidor del sistema. De esta manera, sin una conexión a internet será imposible realizar el envío de información desde el dispositivo móvil al servidor, con la consecuente imposibilidad de almacenar los registros en la base de datos.
- **Permisos de usuario:** dada la tipología de los datos que se van a recoger, en relación básicamente al posicionamiento del usuario, y dada la necesidad de acceder a los recursos nativos del dispositivo móvil, es indispensable para el funcionamiento de la aplicación que el usuario dé su consentimiento en relación con el uso de ambos aspectos.
- **Limitaciones Hardware:** en lo que al servidor respecta, resulta evidente que el registro de rutas y respuestas depende enteramente del correcto funcionamiento del servidor donde se aloja el sistema, puesto que en caso contrario la base de datos y la propia aplicación serán totalmente inoperativas.

Más aún, se deducen diversas restricciones intrínsecas derivadas de la tipología híbrida de la aplicación. Así, se presupone que su rendimiento será menor al de una aplicación móvil nativa y que, más importante, sus funciones respecto al acceso de los recursos del dispositivo móvil son limitadas. Esto constituye una restricción clave puesto que el funcionamiento y aprovechamiento de la aplicación depende de su acceso a dichos recursos, como pueden ser el posicionamiento o el acelerómetro del dispositivo. El uso Apache Cordova permite disponer de plugins que proveen de funcionalidades nativas para así solventar esta restricción.

Relacionado con este último aspecto, y atendiendo a la propia naturaleza de la aplicación, una condición casi indispensable para que sea funcional es que el usuario mantenga activo en todo momento el posicionamiento del dispositivo móvil.

- Requisitos de lenguaje: los lenguajes a utilizar son los principales lenguajes de programación web, como son básicamente HTML, CSS y JavaScript, además de PHP como lenguaje de servidor para comunicarse con la base de datos.

3.8 Documentación de Usuario

El sistema cuenta con dos manuales de usuario, uno para la aplicación web de escritorio, denominado *Semper ambulans Versió d'escriptori, Manual d'usuari*, y otro para la aplicación web móvil, denominado *Semper ambulans Versió mòbil, Manual d'usuari*. Estos documentos son propiedad del GEMOTT a modo de guía de uso y punto de partida para poder llevar a cabo futuras ampliaciones del sistema.

3.9 Asunciones y dependencias

Para el desarrollo del sistema se asumen dos tipos de dependencias:

- Planificación: se asume que los periodos de tiempo fijados para el desarrollo del sistema están correctamente estimados.
- Elementos externos: se asume que los mecanismos físicos externos del sistema, como pueden ser los dispositivos móviles de los usuarios, los dispositivos de los administradores y el servidor del sistema son completamente funcionales y se pueden comunicar correctamente entre sí.

4 METODOLOGÍA DE TRABAJO

4.1 Propósito del apartado

El propósito del presente apartado es facilitar la información de referencia necesaria a las personas implicadas en el desarrollo del sistema. Así, se pretende dar una descripción de la metodología de trabajo que se llevará a cabo en el desarrollo del proyecto, incluyendo en este aspecto tanto la planificación de este, donde se describen actividades, plazos y reuniones con los clientes, entre otros, como diferentes análisis relativos al propio proyecto, tal como puede ser el análisis de riesgos o análisis de alcance asociados a las metodologías empleadas.

4.2 Contexto de la metodología

El presente apartado describe la implementación de una aproximación de las metodologías de trabajo *ágiles* para la gestión del desarrollo del proyecto *Semper ambulans*. Es importante resaltar que debido tanto a la falta de experiencia como a la restricción imperativa que resulta ser el poco tiempo para llevar a cabo el proyecto, además de que el desarrollo del sistema está a cargo de una única persona, se intentará llevar a cabo una implementación de este tipo de metodología lo más aproximada posible a la teoría y buenas prácticas que plantea, aunque será imposible implementarla de manera completa.

Así, en este apartado se incluye, junto con la descripción del ciclo de vida iterativo e incremental para desarrollar el proyecto, los artefactos o documentos con los que se gestionan las tareas de adquisición y suministro, tales como monitorización y seguimiento del avance, así como las responsabilidades y compromisos de los participantes en el proyecto. El análisis de requisitos del proyecto no se incluye en este apartado, sino que habrá que ver el apartado 5.

4.3 Alcance

Se espera que el público objetivo de este apartado sean aquellas personas implicadas en el desarrollo del sistema, como pueden ser el desarrollador, su tutor y los investigadores, como clientes finales del sistema.

4.4 Metodología

4.4.1 Fundamentos

El método de trabajo empleado consiste en una aproximación de metodología ágil de gestión de proyectos de software, esto es, una metodología donde los requisitos y soluciones evolucionan con el tiempo según la necesidad del proyecto. Se caracteriza por adoptar una estrategia de desarrollo incremental, en lugar de planificación y ejecución completa del producto, y el solapamiento de las diferentes fases del desarrollo, en lugar de desarrollar una tras otra en un ciclo secuencial.

Las principales razones del uso de un ciclo de desarrollo iterativo e incremental para la ejecución de este proyecto son:

- Sistema modular: las características del sistema permiten desarrollar una base funcional mínima y sobre ella ir incrementando las funcionalidades o modificando el comportamiento o apariencia de las ya implementadas.
- Entregas frecuentes y continuas al cliente de los módulos terminados, de forma que puede disponer de una funcionalidad básica en un tiempo mínimo y a partir de ahí un incremento y mejora continua del sistema.
- Previsible inestabilidad de requisitos:
 - Es posible que el sistema incorpore más funcionalidades de las inicialmente identificadas.
 - Es posible que durante la ejecución del proyecto se altere el orden en el que se desean recibir los módulos o historias de usuario terminadas.
 - Para el equipo de investigadores resulta difícil precisar cuál será la dimensión completa del sistema, y su crecimiento puede continuarse en el tiempo suspenderse o detenerse.

4.4.2 Responsabilidades

Tutor

- Supervisión de la pila de producto.
- Supervisión y asesoramiento del equipo técnico para la subsanación de las deficiencias que observe.
- Comunicación con el equipo de investigadores para pedir aclaración de las dudas que pueda tener.

Equipo de investigadores

- Registro en la lista de pila del producto de los componentes que definen el sistema.
- Recepción del producto o presentación de reparos.
- Comunicación con el tutor y el equipo técnico para aclarar dudas y subsanar deficiencias.

Equipo técnico

- Conocimiento y comprensión actualizados de la pila de producto.
- Resolución de dudas o comunicación de sugerencias con el tutor y el equipo de investigadores.
- Elaboración del análisis de requisitos, diseño de la base de datos del sistema, diseño de la interfaz de los clientes e implementación del sistema.
- Presentación de incrementos.
- Presentación del producto final, no sobre simulaciones o imágenes.

4.4.3 Planificación

La planificación inicial del proyecto plantea que se inicie el día 8 de abril de 2019 y se finalice el día 12 de julio, con la entrega del producto finalizado y utilizable. Así, se han definido un total de 21 actividades diferentes englobadas en 5 módulos de desarrollo, donde se le ha asignado un identificador a cada actividad y módulo y se ha identificado la secuencia lógica o flujo de trabajo del proyecto. Así queda contemplado en la tabla 2 (ver abajo).

ID	Actividad	Fecha de inicio	Fecha final	Predecesora
A	Especificar requisitos y evaluar proyecto	08/04/2019	30/04/2019	-
A-I	Elaboración del cronograma	08/04/2019	08/04/2019	-
A-II	Recolección de requisitos	09/04/2019	28/04/2019	A-I
A-III	Especificación preliminar del software	15/04/2019	28/04/2019	A-I
A-IV	Revisión de especificaciones	29/04/2019	29/04/2019	A-II, A-III
A-V	Incorporar mejoras en requisitos	29/04/2019	30/04/2019	A-IV
B	Análisis y diseño	01/05/2019	08/05/2019	A
B-I	Diseño de la base de datos	01/05/2019	03/05/2019	A-IV
B-II	Diseño de la interfaz de la aplicación móvil	01/05/2019	03/05/2019	A-IV
B-III	Diseño de las funcionalidades de las herramientas a implementar	03/05/2019	05/05/2019	B-I, B-II
B-IV	Desarrollo de prototipos	03/05/2019	05/05/2019	B-I, B-II
B-V	Revisión de diseños	06/05/2019	06/05/2019	B-III
B-VI	Incorporar mejoras en diseño	06/05/2019	08/05/2019	B-IV

C	Implementación	09/05/2019	17/06/2019	B
C-I	Preparación de la estructura de directorios	09/05/2019	10/05/2019	B-V
	Preparación del entorno de librerías			
	Codificación de la estructura HTML del total de vistas de la aplicación			
C-II	Implementación de la base de datos	09/05/2019	10/05/2019	B-V
C-III	Codificación del mapa	11/05/2019	14/05/2019	C-I, C-II
	Codificación de las herramientas de navegación			
	Codificación del control topológico			
C-IV	Revisión de implementaciones	15/05/2019	15/05/2019	C-III
C-V	Codificación de los formularios	16/05/2019	27/05/2019	C-IV
	Codificación del almacenamiento de la información			
C-VI	Codificación del pintado de datos en el mapa de la aplicación móvil	28/05/2019	02/06/2019	C-V
C-VII	Revisión de implementaciones	03/06/2019	03/06/2019	C-VI
C-VIII	Codificación del mapa de la aplicación web	03/06/2019	16/06/2019	C-VII
	Codificación del pintado de datos en el mapa de la aplicación web			
	Codificación de la descarga de datos a través de la aplicación web			
C-IX	Revisión de implementaciones	06/06/2019	06/06/2019	C-VI
C-X	Revisión de implementaciones	17/06/2019	17/06/2019	C-VIII
D	Pruebas unitarias y de integración	17/06/2019	09/07/2019	C
D-I	Estudiar módulos de código	17/06/2019	26/06/2019	C-VI
D-II	Probar módulos y sus componentes	20/06/2019	30/06/2019	D-I
D-III	Detectar fallos	01/07/2019	02/07/2019	D-II
D-IV	Corregir fallos	03/07/2019	07/07/2019	D-III
D-V	Volver a probar código corregido	08/07/2019	10/07/2019	D-IV
D-VI	Revisión del producto	11/07/2019	11/07/2019	D-V
E	Entrega	12/07/2019	12/07/2019	D
E-I	Entrega del producto	12/07/2019	12/07/2019	D-VI

Tabla 2. Planificación de las actividades a llevar a cabo en el proyecto. Fuente: elaboración propia.

4.4.4 Pila de producto

La pila de producto está compuesta por la lista ordenada de todo aquello que el cliente, en este caso el equipo de investigadores del GEMOTT, cree que necesita el producto en un momento inicial. Para determinar la prioridad de cada caso se ha utilizado la técnica de priorización MoSCoW ver apartado 1.5 para ver el significado de cada caso -. Así, la pila de producto queda definida tal y como se muestra en la tabla 3 (ver abajo).

ID	Prioridad	Descripción
1	M	Se deben geolocalizar las rutas que recorren los usuarios
2	M	Deben aparecer formularios cuando el usuario haga una pausa o entre en zonas de interés
3	M	Se debe enviar y almacenar la información en una base de datos de manera segura.
4	M	Se debe poder descargar la información almacenada en la base de datos

5	S	Se debe activar automáticamente el posicionamiento del usuario cuando este se comience a desplazar.
6	C	La aplicación se debe ejecutar en segundo plano.
7	C	El equipo de investigadores puede disponer de una aplicación de <i>back-office</i> consistente en un visor web de escritorio donde se pueda ver la información almacenada en la base de datos.

Tabla 3. Pila de producto del sistema desarrollado. Fuente: elaboración propia.

5 ANÁLISIS DE REQUISITOS

5.1 Propósito del apartado

Dentro de la totalidad del presente documento, este apartado pretende hacer una descripción general, definir casos de uso y especificar y analizar los requisitos, tanto funcionales como no-funcionales, del sistema. En este contexto se entienden tanto la aplicación móvil como la aplicación web. En definitiva, en este apartado se detallan las demandas propuestas por el cliente, en este caso el GEMOTT, instauradas de mutuo acuerdo con el desarrollador. Se trata, por ende, del punto de partida tanto para el cliente como para el equipo técnico.

5.2 Alcance

Se espera que el público objetivo de este apartado sea cualquier persona familiarizada con el desarrollo de proyectos de ingeniería de software.

5.3 Características del sistema

El sistema propuesto debe contar con dos partes principales y bien diferenciadas: una parte de servidor (*back-end*) y una parte de cliente (*front-end*).

En lo que a ella respecta, la parte de servidor estará constituida por una base de datos de tipo relacional en la cual se almacenarán todos los registros referentes a las rutas geolocalizadas de los usuarios y las respuestas a los formularios resueltos por los mismos. El SGBD será de tipo PostgreSQL, el acceso a los datos por parte de los administradores e investigadores se dará a través de PgAdmin4 y el lenguaje de servidor utilizado para comunicarse con la base de datos es PHP. Por lo que concierne a dicho servidor, estará alojado en las instalaciones del LIGIT *Laboratori d'Informació Geogràfica i Teledetecció* -, en la misma Universitat Autònoma de Barcelona.

La parte de cliente estará constituida por dos aplicaciones de tecnología web:

- Por un lado, encontramos una aplicación móvil híbrida destinada exclusivamente a los usuarios de a pie y que constituye el núcleo funcional del sistema. Dicha aplicación será empaquetada mediante la plataforma Phone Gap, un paquete de librerías que convierte los archivos web en un fichero de tipo *apk* instalable en los dispositivos móviles. Esto dota a la aplicación tanto de componentes visualmente nativos como, mucho más importante, de funcionalidades nativas, gracias a que Phone Gap no es más que un producto derivado de Apache Cordova, o más técnicamente, una distribución. Por su lado, Apache Cordova es un entorno de desarrollo de aplicaciones móviles que permite disponer de multitud de plugins que proveen funcionalidades nativas y las cuales son indispensables para el funcionamiento de la aplicación. Así mismo, se utilizará la biblioteca multiplataforma jQuery, tanto para simplificar la manipulación del árbol DOM como para integrar la técnica AJAX, y el *framework* jQuery Mobile, para facilitar la adaptación del sistema

a los dispositivos táctiles como son *smartphones* y *tablets*. Todas las tecnologías mencionadas se explicarán más en profundidad en el apartado 8.

Más aún, y en concreto a las funcionalidades de la aplicación móvil, se espera que esta sea capaz de acceder a dos recursos concretos del dispositivo. Por un lado se debe permitir el posicionamiento del dispositivo a través del receptor GNSS del móvil mientras que, por el otro lado, se debe permitir detectar el movimiento del usuario a través del acelerómetro de su dispositivo, evento que desencadena el inicio de la geolocalización de su ruta. Partiendo de este punto, existen dos escenarios en los cuales se despliega un formulario:

- a) El usuario cruza o entra en uno de los polígonos que componen una capa integrada en la aplicación, de formato *geojson*, en la cual se recogen diferentes entidades territoriales de interés como parques, plazas o playas.
 - b) El usuario se para durante 30 segundos, lo cual es detectado por el acelerómetro.
- Por el otro lado encontramos una aplicación web destinada exclusivamente a los investigadores y administradores, a modo de herramienta de *back-office* de monitorización y explotación.

Para la creación tanto del visor web de la aplicación de *back-office* como del mapa dinámico de la aplicación móvil se utilizará Leaflet, una biblioteca de JavaScript utilizada para implementar mapas interactivos en los navegadores web. Así mismo, y en lo que a funcionalidad respecta, se utilizarán diversas bibliotecas y plugins complementarios, los cuales serán esenciales para que el sistema funcione correctamente y sea capaz de desarrollar los procesos demandados. El total de paquetes funcionales externos queda recogido en la tabla 4.

Nombre	Cliente	Función
Locate	Ambos	Geolocalizar el usuario.
Turf.js	Ambos	Realizar análisis geoespaciales en la web. En el caso concreto del proyecto, detectar cuando el usuario ha entrado en un polígono de interés y calcular la distancia recorrida en una ruta
FileSaver	Escritorio	Permitir la creación y descarga de archivos desde el lado de cliente. En el caso del proyecto, facultar la descarga de datos en formato <i>geojson</i> .

Tabla 4. Definición y funciones de los paquetes externos empleado en el desarrollo del sistema. Fuente: elaboración propia.

5.4 Diagrama de flujo de procesos del sistema

Un diagrama de flujo es un diagrama que describe un proceso, sistema o algoritmo informático. Se usan ampliamente en numerosos campos para documentar, estudiar, planificar, mejorar y comunicar procesos complejos en diagramas claros y fáciles de comprender.

Así, y teniendo esto en cuenta, se ha considerado muy útil la idea de diseñar un diagrama de flujo de procesos para el presente proyecto, si bien se ha decidido recoger solamente el proceso de registro y explotación de información pues es el que se ha considerado más complejo y a la vez más importante del total de flujos del sistema. A tal efecto se ha hecho uso de la aplicación web *Draw.io* (<https://www.draw.io/>), una herramienta web que permite desarrollar y compartir diseños, esquemas y diagramas de manera fácil e intuitiva. De esta manera, el diagrama de flujo de proceso se comprende en la figura 2 - ver siguiente página -.

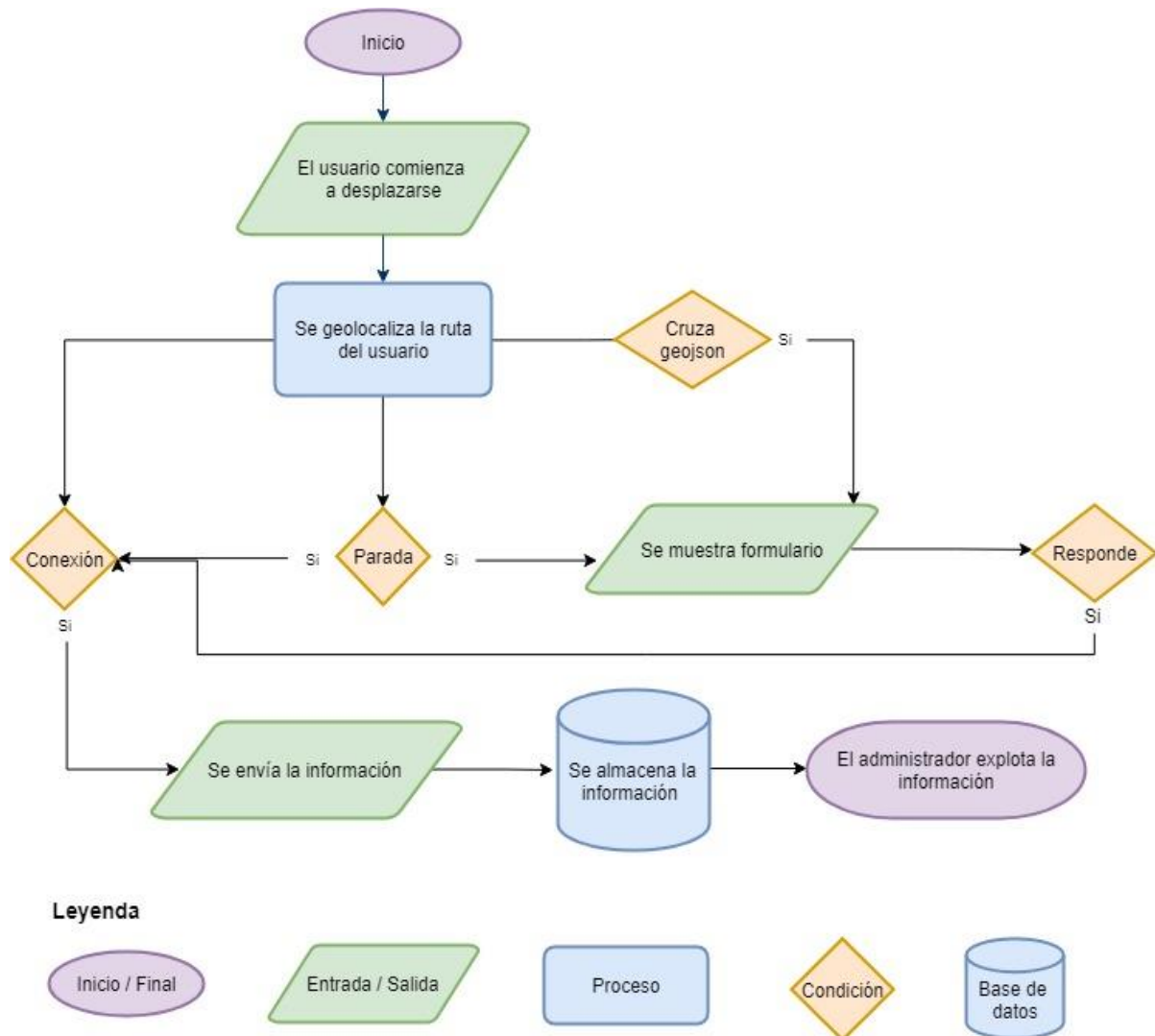
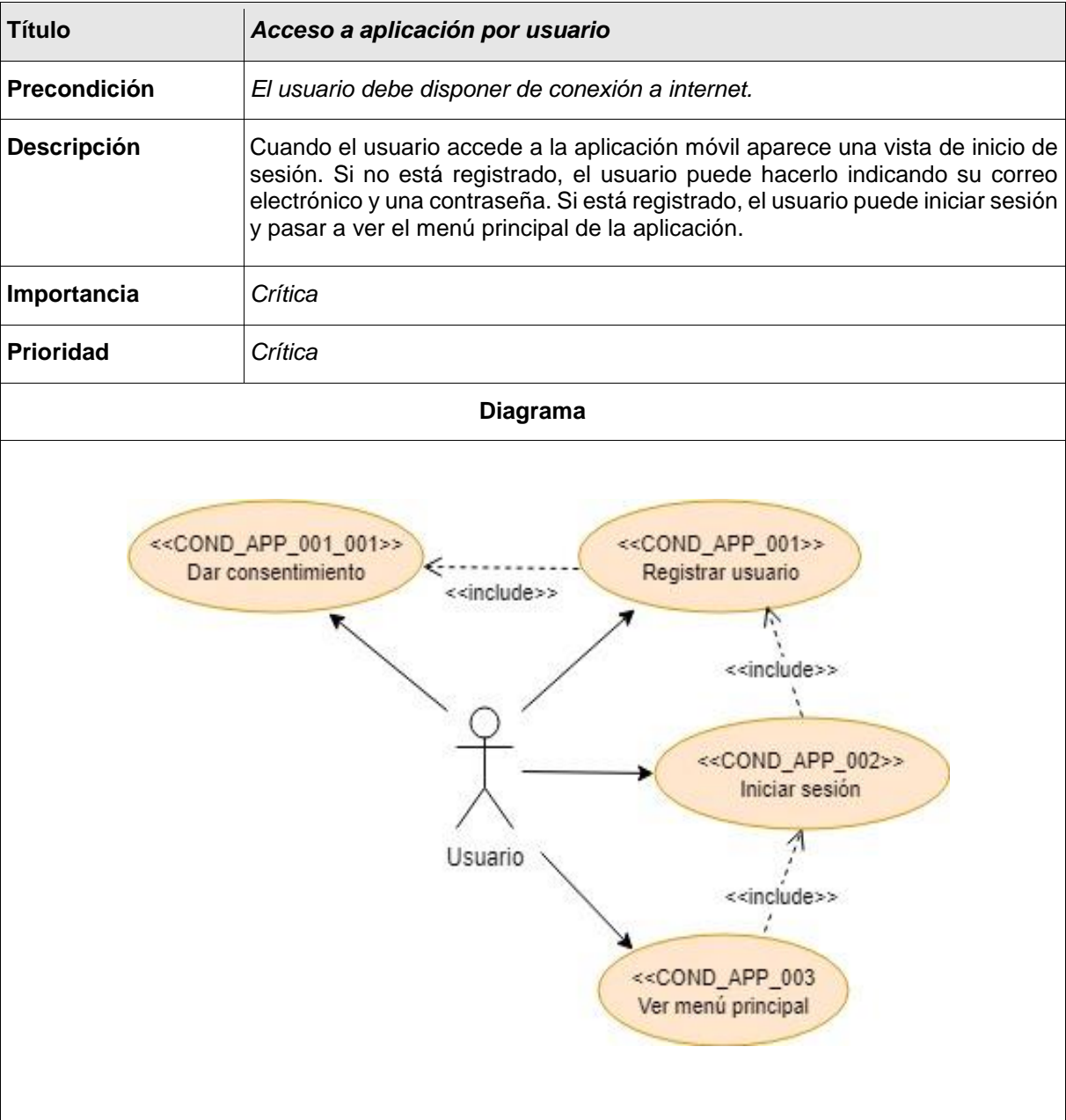


Figura 2. Diagrama de flujo del proceso principal de recogida y explotación de información del sistema desarrollado.
Fuente: elaboración propia.

5.5 Casos de uso del sistema

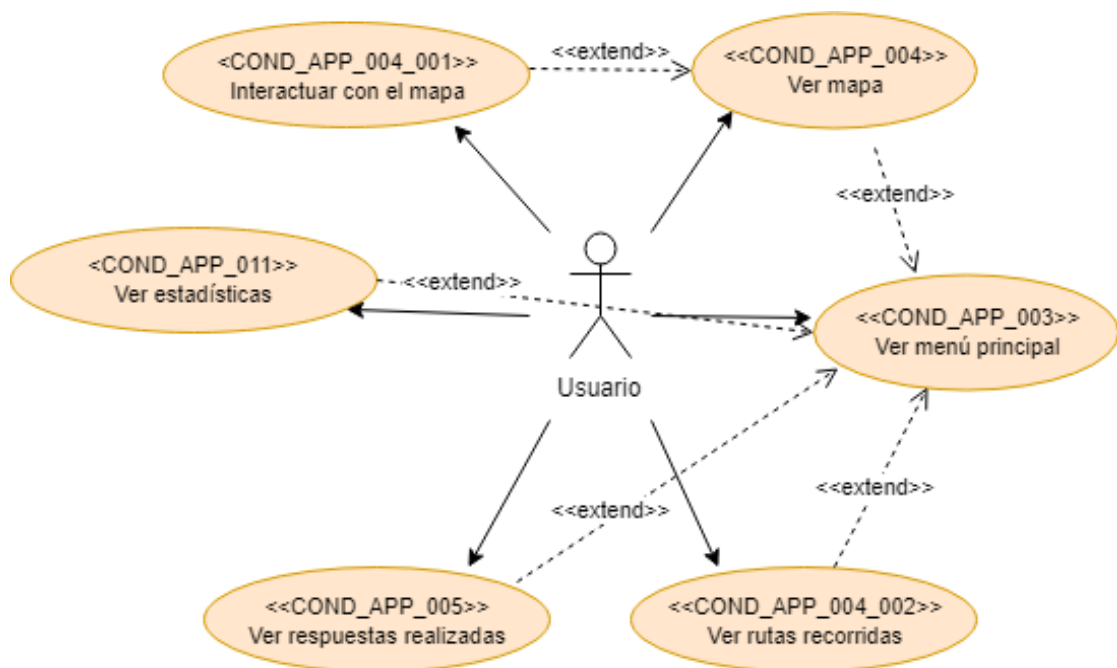
Las características del sistema se especifican organizadas en casos de uso. A continuación, en el apartado 5.5.1, se muestra la totalidad de casos de uso identificados a través de un índice visual, desarrollados todos también con la aplicación web *Draw.io*. A su vez, se definen las especificaciones de cada caso de uso identificado, así como las relaciones internas que se desarrollan tanto entre casos como con los actores del sistema y especificando qué interacción actor-software provoca la ejecución de cada caso de uso. Para ver los requisitos que se señalan en las especificaciones, ver el apartado 5.6. Nótese que la especificación del flujo principal se muestra en una versión extendida respecto a las demás especificaciones, donde vista la complejidad e importancia de este caso de uso se ha considerado necesario llevar a cabo una descripción más precisa y extendida del conjunto de eventos y procesos que se desarrollan.

5.5.1 Definición de los casos de uso del sistema



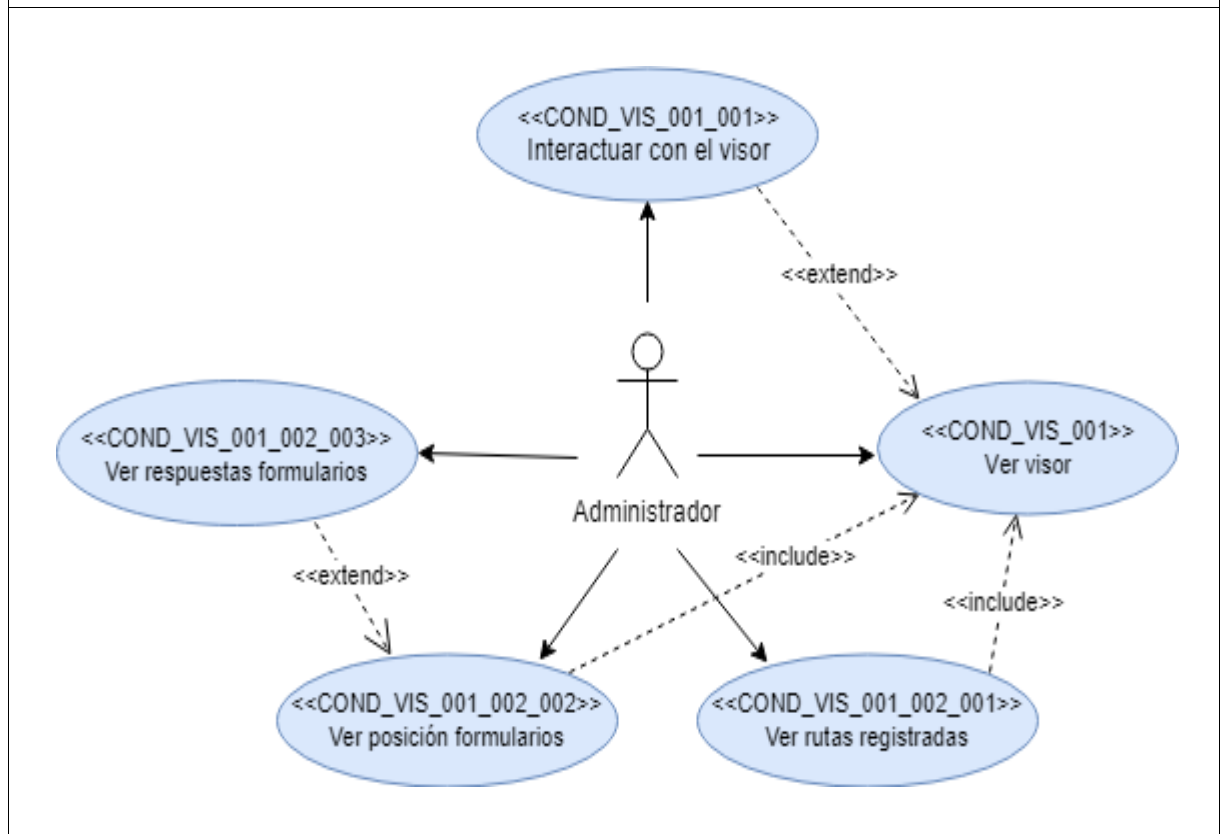
Título	Consulta de información por usuario
Precondición	<i>El usuario debe disponer de conexión a internet y haber iniciado sesión previamente.</i>
Descripción	El usuario puede acceder al menú principal de la aplicación e interactuar con las funcionalidades de cliente de esta. De esta manera, el usuario es capaz de ver sus estadísticas y tanto de abrir como de interactuar con un mapa, donde puede ver las rutas que ha recorrido y los formularios que ha respondido.
Importancia	<i>Media</i>
Prioridad	<i>Baja</i>

Diagrama



Título	Explotación de información por administrador
Precondición	<i>El administrador debe de tener conexión a internet y permisos para acceder a la base de datos. Además, es necesario que haya registros almacenados en la misma para poder explotar dicha información.</i>
Descripción	El administrador puede acceder a un visor de mapas web de escritorio donde es capaz de visualizar toda la información almacenada en la base de datos, ya sean rutas o sean formularios. Además, debe ser capaz de descargar dicha información de la base de datos.
Importancia	<i>Media</i>
Prioridad	<i>Alta</i>

Diagrama



Título	Flujo principal			
Precondición	El usuario debe disponer de conexión a internet en su dispositivo. Además, debe tener activado el posicionamiento de su dispositivo móvil y debe haber iniciado sesión en la aplicación.			
Descripción	El usuario, mediante la aplicación móvil, es capaz tanto de localizar sus rutas a pie, cuando el dispositivo detecta movimiento, como de responder un formulario, cuando ocurren ciertos eventos. Esta información se envía cuando el usuario se detiene de manera prolongada o cuando así lo decide, y se registra en una base de datos.			
Secuencia Normal	Paso	Acción		
	1	El usuario se desplaza		
	2	El dispositivo móvil detecta el movimiento <COND_APP_006_001>		
	3	La aplicación comienza a almacenar la ruta geolocalizada del usuario <COND_APP_008>		
	4.1	Si el usuario cruza un polígono determinado <COND_APP_007_002>		
		Aparece un formulario <COND_APP_009>		
		El usuario responde el formulario <COND_APP_009_001>		
		Se guardan las respuestas y se asignan a la ruta <COND_APP_009_001_001>		
	4.2	Si el usuario se detiene durante 30 segundos <COND_APP_006_002>		
		Aparece un diálogo de decisión <COND_APP_006_002_001>		
		Aparece un formulario <COND_APP_009>	Se envía la ruta realizada <COND_APP_008_001>	El usuario continúa caminando
		El usuario responde el formulario <COND_APP_009_001>		
		Se guardan las respuestas y se asignan a la ruta <COND_APP_009_001_001>		
Importancia	Crítica			
Prioridad	Crítica			
Diagrama	Ver página siguiente			



5.6 Requisitos funcionales

5.6.1 Requisitos de información

ID	INF_001
Título	Almacenar datos de usuario
Descripción	El sistema debe ser capaz de almacenar en la base de datos las credenciales de inicio de sesión de cada usuario. Dichas credenciales son su correo electrónico y una contraseña.
Requisito padre	-

ID	INF_002
Título	Almacenar respuestas
Descripción	El sistema debe ser capaz de almacenar en la base de datos las respuestas de los usuarios a los formularios respondidos.
Requisito padre	-

ID	INF_003
Título	Almacenar rutas geolocalizadas
Descripción	El sistema debe ser capaz de almacenar en la base de datos las rutas geolocalizadas que recorren los usuarios.
Requisito padre	-

5.6.2 Requisitos de conducta

Aplicación móvil

ID	COND_APP_001
Título	Registrar usuario
Descripción	La aplicación debe permitir al usuario registrarse en el sistema indicando su correo electrónico y una contraseña.
Interfaz de servicio	Si
Requisito padre	-

ID	COND_APP_001_001
Título	Dar consentimiento
Descripción	La aplicación debe pedir al usuario que dé su consentimiento en términos de compartir su ubicación y la información derivada de ella a los investigadores del proyecto. Esto solo se da una vez, cuando el usuario se registra.
Interfaz de servicio	Si
Requisito padre	-

ID	COND_APP_002
Título	Iniciar sesión
Descripción	La aplicación debe permitir al usuario iniciar sesión indicando su correo electrónico y su contraseña.
Interfaz de servicio	Si
Requisito padre	-

ID	COND_APP_003
Título	Ver menú principal
Descripción	La aplicación debe permitir al usuario debe poder ver un menú principal a través del cual puede acceder a otras vistas.
Interfaz de servicio	Si
Requisito padre	-

ID	COND_APP_004
Título	Ver mapa
Descripción	La aplicación debe permitir al usuario ver e interactuar con un mapa de la ciudad de Barcelona.
Interfaz de servicio	Si
Requisito padre	-

ID	COND_APP_004_001
Título	Interactuar con el mapa
Descripción	La aplicación debe permitir al usuario interactuar con un mapa de Barcelona, a través de acciones como moverse por el mapa (<i>pan</i>), alejarse (<i>zoom out</i>) o acercarse (<i>zoom in</i>).
Interfaz de servicio	Si
Requisito padre	COND_APP_004

ID	COND_APP_004_002
Título	Ver rutas recorridas
Descripción	La aplicación debe permitir al usuario ver, de manera geolocalizada, las rutas que ha recorrido y que se han registrado utilizando la aplicación.
Interfaz de servicio	Si
Requisito padre	COND_APP_004

ID	COND_APP_005
Título	Ver respuestas realizadas
Descripción	La aplicación debe permitir al usuario ver, de manera geolocalizada, las respuestas a los formularios que ha respondido con anterioridad.
Interfaz de servicio	Si
Requisito padre	-

ID	COND_APP_006
Título	Acceder al acelerómetro.
Descripción	La aplicación debe ser capaz de acceder a las funciones del acelerómetro del dispositivo móvil.
Interfaz de servicio	No
Requisito padre	-

ID	COND_APP_006_001
Título	Detectar movimiento
Descripción	La aplicación debe ser capaz de detectar el movimiento del usuario.
Interfaz de servicio	No
Requisito padre	COND_APP_006

ID	COND_APP_006_002
Título	Detectar pausa
Descripción	La aplicación debe ser capaz de detectar una pausa de 30 segundos por parte del usuario.
Interfaz de servicio	No
Requisito padre	COND_APP_006

ID	COND_APP_006_002_001
Título	Mostrar diálogo de decisión
Descripción	La aplicación debe mostrar un diálogo de decisión cuando detecte una pausa de 30 segundos.
Interfaz de servicio	Si
Requisito padre	COND_APP_006_002

ID	COND_APP_007
Título	Activar localización
Descripción	La aplicación debe ser capaz de activar el posicionamiento GNSS del dispositivo móvil cuando ha detectado el movimiento del usuario.
Interfaz de servicio	No

Requisito padre	-
------------------------	---

ID	COND_APP_007_001
Título	Geolocalizar usuario
Descripción	La aplicación debe ser capaz de conocer la posición del dispositivo móvil.
Interfaz de servicio	No
Requisito padre	COND_APP_007

ID	COND_APP_007_002
Título	Detectar cruce con geojson
Descripción	La aplicación debe ser capaz de detectar cuando el usuario entra o cruza alguno de los polígonos que forman la capa <i>geojson</i> de zonas de interés.
Interfaz de servicio	No
Requisito padre	COND_APP_007

ID	COND_APP_007_003
Título	Posicionar usuario
Descripción	La aplicación debe ser capaz de mostrar al usuario la posición del dispositivo móvil dentro del mapa.
Interfaz de servicio	No
Requisito padre	COND_APP_007

ID	COND_APP_008
Título	Geolocalizar ruta
Descripción	La aplicación debe ser capaz de geolocalizar las rutas que recorren los usuarios.
Interfaz de servicio	No
Requisito padre	-

ID	COND_APP_008_001
Título	Enviar ruta geolocalizada
Descripción	La aplicación debe ser capaz enviar las rutas geolocalizadas al servidor una vez ha detectado una parada de 30 segundos.
Interfaz de servicio	No
Requisito padre	COND_APP_008

ID	COND_APP_009
Título	Mostrar formulario
Descripción	La aplicación debe ser capaz de mostrar un formulario al usuario cuando este entra o cruza alguno de los polígonos que forman la capa <i>geojson</i> de zonas de interés o realiza una pausa de 30 segundos.
Interfaz de servicio	Si
Requisito padre	-

ID	COND_APP_009_001
Título	Responder formulario
Descripción	La aplicación debe permitir al usuario responder las preguntas de los formularios mostrados.
Interfaz de servicio	Si
Requisito padre	COND_APP_009_001

ID	COND_APP_009_001_001
Título	Enviar respuestas
Descripción	La aplicación debe ser capaz de enviar las respuestas a los formularios por parte del usuario.
Interfaz de servicio	No
Requisito padre	COND_APP_009_001

ID	COND_APP_010
Título	Detectar conexión a internet
Descripción	La aplicación debe ser capaz de detectar si el dispositivo móvil dispone de conexión a internet.
Interfaz de servicio	No
Requisito padre	-

ID	COND_APP_011
Título	Ver estadísticas
Descripción	La aplicación debe permitir al usuario ver sus estadísticas en términos de rutas registradas, formularios respondidos y distancia total recorrida.
Interfaz de servicio	Si
Requisito padre	-

Aplicación web

ID	COND_VIS_001
Título	Ver visor
Descripción	El visor debe permitir al administrador ver e interactuar con un mapa de la ciudad de Barcelona.
Interfaz de servicio	Si
Requisito padre	-

ID	COND_VIS_001_001
Título	Interactuar con el visor
Descripción	El visor debe permitir al administrador interactuar con un mapa de Barcelona, a través de acciones como moverse por el mapa (<i>pan</i>), alejarse (<i>zoom out</i>) o acercarse (<i>zoom in</i>).
Interfaz de servicio	Si
Requisito padre	FUNC_COND_VIS_001

ID	COND_VIS_001_002
Título	Ver información almacenada
Descripción	El visor debe permitir al administrador ver sobre el mapa de la ciudad de Barcelona la información almacenada en la base de datos.
Interfaz de servicio	Si
Requisito padre	FUNC_COND_VIS_001

ID	COND_VIS_001_002_001
Título	Ver rutas registradas
Descripción	El visor debe permitir al administrador ver las rutas geolocalizadas recorridas por los usuarios.
Interfaz de servicio	Si
Requisito padre	FUNC_COND_VIS_001_002

ID	COND_VIS_001_002_002
Título	Ver formularios registrados
Descripción	El visor debe permitir al administrador ver la posición de los formularios respondidos por los usuarios en cada ruta.
Interfaz de servicio	Si
Requisito padre	FUNC_COND_VIS_001

ID	COND_VIS_001_002_003
Título	Ver respuestas formularios
Descripción	El visor debe permitir al administrador hacer clic sobre la posición de un formulario y mostrar las respuestas del usuario en una ventana emergente (<i>pop-up</i>).
Interfaz de servicio	Si
Requisito padre	FUNC_COND_VIS_001_002

5.7 Requisitos no funcionales

5.7.1 Requisitos de fiabilidad

ID	FIAB_001
Título	Tolerancia a errores
Descripción	El porcentaje de procesos u acciones erróneas que realiza el sistema respecto al total de acciones no debe ser superior al 1%.

5.7.2 Requisitos de usabilidad

ID	USAB_001
Título	Facilidad de aprendizaje
Descripción	La aplicación y el visor deben mostrar un diseño intuitivo y amigable que permita a usuarios y administradores familiarizarse y explotar al máximo el sistema en un tiempo no superior a 1 día.

5.7.3 Requisitos de mantenibilidad

ID	MANT_001
Título	Facilidad de ampliación
Descripción	El sistema debe estar diseñado y pensado para que pueda ser ampliado fácilmente, en términos de actualizaciones futuras y contemplando la posibilidad de añadir nuevos módulos de distintas funcionalidades tanto a la aplicación como al visor.

ID	MANT_002
Título	Facilidad de testeo
Descripción	La aplicación y el visor deben estar diseñados y pensados para que puedan ser sometidos de manera sencilla a un proceso de testeo de sus componentes y funciones.

ID	MANT_003
Título	Escalabilidad
Descripción	La aplicación y el visor deben estar diseñados y pensados para que no pierdan rendimiento conforme aumente su público.

5.7.4 Requisitos de portabilidad

ID	PORT_001
Título	Instalación aplicación
Descripción	La aplicación debe ser descargable y accesible a través del repositorio de aplicaciones del correspondiente sistema operativo.

ID	PORT_002
Título	Acceso a internet
Descripción	La aplicación debe ser capaz de acceder a internet sin ningún tipo de restricción o control.

ID	PORT_003
Título	Actualización aplicación
Descripción	La aplicación debe ser actualizable, y descargar e instalar los paquetes de actualización que salgan al mercado

ID	PORT_004
Título	Compatibilidad con el dispositivo móvil
Descripción	La aplicación debe ser ejecutable únicamente con el software de fábrica del dispositivo móvil, sin ser necesaria la descarga de recursos externos.

5.7.5 Requisitos de seguridad

ID	SEG_001
Título	Privacidad de los usuarios
Descripción	La aplicación debe mantener el anonimato y privacidad de los usuarios de la misma.

ID	SEG_002
Título	Acceso a la base de datos
Descripción	Sólo los administradores pueden acceder a la base de datos del sistema y deben disponer de permisos para ello.

5.7.6 Requisitos técnicos

ID	TEC_001
Título	Almacenamiento local de datos

Descripción	Los datos referentes a las ruta que se esté recorriendo actualmente se almacenarán localmente en dicho dispositivo en el correspondientes archivo de formato JSON.
--------------------	--

ID	TEC_002
Título	Sistema de Gestión de la Base de datos
Descripción	La base de datos del sistema será gestionada a través de un SGBD de tipo PostgreSQL.
Versión	9.3

ID	TEC_003
Título	Administración de la base de datos
Descripción	El SGBD de tipo PostgreSQL será administrado y manejado a través de la herramienta PgAdmin4, a la cual se accederá a través de un navegador web y conexión a internet.
Versión	1.0

ID	TEC_004
Título	Sistema operativo
Descripción	La aplicación móvil se desarrollará para el sistema operativo móvil Android.
Versión	-

ID	TEC_005
Título	Entorno de desarrollo
Descripción	La aplicación móvil se desarrollará utilizando el SDK Ionic.
Versión	-

5.7.7 Requisitos de integración

ID	INT_001
Título	Biblioteca Leaflet
Descripción	La aplicación y el visor deberán utilizar la biblioteca de JavaScript Leaflet para implementar los mapas de la Ciudad de Barcelona.
Versión	1.5.1

ID	INT_001_001
Título	Plugin Leaflet Locate
Descripción	La aplicación deberá utilizar el plugin de Leaflet Leaflet.Locate para poder localizar el dispositivo móvil dentro del mapa de la aplicación
Versión	0.66.2

ID	INT_002
Título	Biblioteca Turf
Descripción	La aplicación deberá utilizar la biblioteca para JavaScript <i>Turf.js</i> para poder realizar el análisis topológico concerniente a la intersección de un punto con un polígono.
Versión	-

ID	INT_003
Título	Plugin Accelerometer
Descripción	La aplicación deberá utilizar el plugin de Cordova <i>Accelerometer</i> para poder acceder a las funciones del acelerómetro del dispositivo móvil y así detectar el movimiento del mismo.
Versión	-

ID	INT_004
Título	Plugin Geolocation
Descripción	La aplicación deberá utilizar el plugin de Cordova <i>Geolocation</i> para poder acceder al posicionamiento GNSS del dispositivo móvil y así posicionar al mismo.
Versión	-

ID	INT_005
Título	Biblioteca JQuery
Descripción	La aplicación deberá utilizar la biblioteca JQuery para simplificar la interacción con los documentos HTML, manipular el árbol DOM y agregar interacción con la técnica AJAX.
Versión	2.2.4

ID	INT_006
Título	Framework JQuery mobile
Descripción	La aplicación deberá utilizar el framework JQuery Mobile con el objetivo de simplificar el proceso de adecuar la aplicación a dispositivos táctiles.
Versión	1.4.5

ID	INT_007
Título	Tema NativeDroid ²
Descripción	La aplicación utilizará el tema de JQuery Mobile NativeDroid ² con el objetivo de implementar, de manera sencilla, una interfaz de usuario amigable y agradable.
Versión	2.0.1

ID	INT_008
Título	Gestor de paquetes web Bower
Descripción	La aplicación utilizará el gestor de paquetes web Bower con el objetivo de instalar el tema NativeDroid ² .
Versión	1.8.8

ID	INT_009
Título	Gestor de paquetes npm
Descripción	La aplicación utilizará el gestor de paquetes web <i>npm</i> con el objetivo de instalar tanto los paquetes que sean necesarios para el correcto funcionamiento de la aplicación como sus dependencias.
Versión	5.1.0

ID	INT_009_001
Título	Entorno Node.js
Descripción	La aplicación tendrá instalado el entorno de ejecución para JavaScript <i>Node.js</i> , con el objetivo de utilizar su gestor de paquetes <i>npm</i> .
Versión	10.16.0

6 ANÁLISIS DEL SISTEMA

6.1 Propósito del apartado

El objetivo del presente apartado es describir los diversos análisis que se han desarrollado para enfocar y analizar diversos aspectos del proyecto y en definitiva del sistema desarrollado. Básicamente se han llevado a cabo dos tipos de análisis distintos, como son un análisis de alcance, o *DAFO* *Debilidades, Amenazas, Fortalezas y Oportunidades* y un análisis de riesgos. El objetivo de este proceso es esencialmente detectar tanto las debilidades como las fortalezas, en pos (1) de poder anticiparnos a los riesgos a los que se enfrenta el sistema, por un lado, y (2) poder explotar y maximizar sus fortalezas, por el otro lado.

6.2 Análisis de alcance

El análisis de alcance, o *DAFO*, es una herramienta de estudio muy utilizada para averiguar la situación real de un proyecto, analizando tanto sus características internas - **D**ebilidades y **F**ortalezas como externas **A**menazas y **O**portunidades en una matriz.

De esta manera, y vista la definición del concepto, parece evidente que este tipo de análisis puede ser de gran utilidad para el desarrollo del producto, pues permite detectar tanto sus debilidades, dotándonos de la capacidad de anticiparnos a ellas y desarrollar métodos de mitigación y anticipación, como sus fortalezas, permitiendo saber en qué aspectos el sistema es potente e inducirnos a maximizar dichos aspectos.

Así, en la figura 3 - ver siguiente página - se contempla la matriz *DAFO* elaborada en pos de llevar a cabo dicho análisis. De entre las diversas conclusiones que se pueden extraer de la matriz, resulta pertinente señalar varias de ellas. En primer lugar, parece claro que el sistema muestra una total dependencia de la conexión a internet, pues de ella depende la comunicación entre los clientes y la base de datos. Así, tanto el envío como la recepción y explotación de la información registrada depende críticamente de una conexión a internet estable. Más aún, el elevado ritmo de innovación y desarrollo de

nuevas tecnologías en el ámbito geoespacial puede resultar ser un arma de doble filo, puesto que es probable que las versiones de las diversas bibliotecas y paquetes externos empleados se queden obsoletos o incluso totalmente inoperativos, volviendo el sistema inutilizable y de poco valor.

No obstante, este último aspecto también se puede considerar una ventaja, puesto que existe un amplio abanico de aplicaciones y sistemas parecidos cuyas experiencias y resultados pueden establecerse como base para futuras ampliaciones y mejoras del sistema. La potencia que le otorga este hecho al proyecto es que se implementará un sistema fácilmente ampliable y actualizable y, lo que es más importante, ampliamente extrapolable a los ámbitos y temáticas que el equipo investigador del GEMOTT considere oportunas. Así, es innegable que la posibilidad de expandir el uso del sistema como medio de obtención de información a otras líneas ya existentes, o incluso desarrollar nuevas líneas en base a la posibilidad de recopilar información haciendo uso del sistema, resultan ser oportunidades de lo más atractivas dentro del ámbito de la investigación científica, donde la obtención de datos puede ser en ocasiones un proceso tedioso.

FACTORES INTERNOS		FACTORES EXTERNOS	
DEBILIDADES (-)		AMENAZAS (-)	
1	Total dependencia de una conexión a internet.	1	Elevado ritmo de innovación en el sector, induciendo a que los métodos empleados se queden obsoletos.
2	Falló tecnológico del dispositivo móvil, como fallos en el receptor GNSS o un acelerómetro mal calibrado.	2	Abandono o desuso de la aplicación por parte de los usuarios.
3	Poca atracción de los usuarios por la aplicación móvil si no existe una campaña publicitaria.	3	Cantidad y tipología de la información recogida irrelevante y/o poco fiable.
4	Aplicación web móvil híbrida y no nativa, con las consecuentes desventajas en cuanto a rendimiento y uso de los recursos del dispositivo.	4	Robo de información respecto a la actividad, datos y opiniones de los usuarios.
5	Probable consumo de recursos del dispositivo móvil elevado, en función del modelo de este.	5	La consistencia de los resultados depende del voluntariado.
FORTALEZAS (+)		OPORTUNIDADES (+)	
1	Amplios antecedentes, con numerosas aplicaciones móviles parecidas en el mercado.	1	Posibilidad de expandir el uso del sistema a otras líneas de investigación ya existentes.
2	Medio muy potente de obtención de datos en vivo.	2	Creación y desarrollo de nuevas líneas de investigación centradas en el uso del sistema.
3	Retorno de información al usuario, para que pueda hacer un seguimiento de sus actividades.	3	Ampliación y actualización del sistema desarrollado.
4	Gratuita.	4	Mínima inversión inicial de tiempo y fondos.
5	No requiere de atención constante por parte del usuario.	5	Posibilidad de monetizar la aplicación móvil mediante anuncios.

Figura 3. Matriz DAFO para análisis estratégico diseñada para el proyecto desarrollado. Fuente: PlanillaExcel.com

6.3 Análisis de riesgos

El análisis de riesgo es un método sistemático de estudio de las causas de las posibles amenazas que pueden alterar y afectar el sistema desarrollado, además de los

daños y consecuencias que estas pueden producir sobre el mismo. Se trata de una herramienta muy potente en proyectos de esta índole, especialmente a la hora de definir y diseñar el proyecto, puesto que permite anticiparnos a los posibles problemas a hacer frente y desarrollar de antemano métodos de mitigación. Si bien es cierto que este tipo de análisis puede llegar a conformar un proceso muy complejo, se ha reducido el desarrollo a dos fases principales:

- i. **Evaluación del riesgo:** en la cual se tiene en cuenta la probabilidad de que se produzca un peligro y las consecuencias si ocurre.
- ii. **Gestión del riesgo:** la cual consiste en la identificación y aplicación de la mayor opción para reducir o eliminar la probabilidad de que se dé un peligro.

De esta manera, y sin más preámbulo, se han recogido el total de riesgos y amenazas a los que es probable que el sistema desarrollado deba hacer frente. Dichos elementos se contemplan en la tabla 5 - ver abajo -, donde no sólo se recogen dichos riesgos si no que también se analiza su probabilidad, impacto y ámbito de afectación, ofreciendo además herramientas para mitigar e incluso evitar dichos riesgos.

Descripción del riesgo	Consecuencia	Probabilidad	Impacto	Mitigación	Afectación
Actualización de las bibliotecas externas utilizadas	Aplicación inoperativa	Baja	Alto	Utilización de bibliotecas de pago o últimas versiones estables	Coste
Ausencia de conexión a internet	Aplicación inoperativa	Media	Alto	Descarga y almacenamiento de datos del servidor en el dispositivo móvil	Entrega
Porcentaje de procesos erróneos mayor al 1% de los procesos realizados	Descontento del cliente y de los usuarios. Mal funcionamiento de las aplicaciones	Baja	Medio	Aumentar el número de las pruebas de calidad	Entrega / Reputación
Fallo de seguridad	Robo de datos de los usuarios. Descontento de los usuarios	Baja	Medio	Mantener el anonimato de los usuarios para no relacionar su información con ellos	Entrega
Falta de precisión de posicionamiento del dispositivo móvil de los usuarios	Información obtenida poco representativa	Media	Alto	Aumentar el número de muestras para asegurar la representatividad de los datos	Resultados

Tabla 5. Registro de riesgos y mitigaciones detectados del sistema desarrollado. Fuente: elaboración propia.

De esta manera, y en consonancia al análisis de alcance desarrollado en el apartado anterior, queda de manifiesto que el sistema tiene que hacer frente a diversos escenarios adversos que se escapan de nuestro alcance, como puede ser esencialmente la falta de conexión a internet, lo cual básicamente depende del usuario y no entra en el alcance del sistema, y la actualización de las bibliotecas externas empleadas, lo cual depende del ritmo de innovación del ámbito y puede tener impacto o no sobre el sistema, en función del tipo de actualización.

7 DISEÑO

7.1 Propósito del apartado

Dentro de la totalidad del presente documento, este apartado pretende hacer una descripción detallada y concisa de los diferentes procesos llevados a cabo en cuanto a diseño se refiere. De esta manera, en este apartado se recoge tanto el desarrollo del prototipado inicial de las interfaces de usuario de la aplicación móvil del sistema - ver apartado 6.2 - como la totalidad del proceso llevado a cabo en relación con la definición y diseño de la base de datos de este último - ver apartado 6.3 -.

7.2 Prototipado de las interfaces de usuario

El prototipado es una fase crítica del diseño y desarrollo de una aplicación web móvil como la propuesta, donde los prototipos ayudan tanto a explorar caminos de concepto o diseño como a detectar posibles problemas o carencias antes de empezar la fase de programación.

Así, en función de su grado de complejidad o semejanza con el diseño final, podemos distinguir entre prototipos de baja, media o alta fidelidad. En el caso concreto del presente proyecto se ha decidido optar por las representaciones de baja fidelidad, también denominados *wireframes*. Su contenido suele limitarse a tres cuestiones básicas, como son los grupos de contenido, la estructura de la información y la descripción y visualización básica de la interacción entre el usuario y la interfaz.

De esta manera, en el momento inicial del prototipado de interfaces de usuario se diseñaron un total de 5 interfaces diferentes, una para cada interfaz de usuario o vista que se dispuso en el momento previo a la implementación del sistema, a saber:

- Vista de registro
- Vista de inicio de sesión
- Menú principal
- Menú principal con menú desplegable
- Vista de consentimiento de derechos de uso

Para el diseño y elaboración de los *wireframes* se hizo uso de la aplicación web *Mockup Builder* (<http://mockupbuilder.com/>), una herramienta web para elaborar, compartir y validar de manera simple y fácil las soluciones de prototipado que se desarrollen. La totalidad de las vistas elaboradas se recogen en las figuras 1 a 5 del anexo.

7.3 Base de datos del sistema

7.3.1 Definición de la base de datos

El objetivo principal a la hora de configurar el conjunto de datos del sistema es conseguir que estén bien estructuradas y que su uso sea el más sencillo y fluido posible, simplificando el trabajo con el sistema y garantizando el éxito de su implementación.

Existen diversos modelos de bases de datos en función de la estructura utilizada a la hora de construirlas. Estos modelos definen la estructura, operadores y reglas para contener, organizar y operar un conjunto de datos. Para el sistema propuesto se ha decidido utilizar el modelo de datos relacional (Codd, 1970), el cual es el más utilizado actualmente dada su simplicidad, la cual le otorga una gran potencia. Utiliza únicamente tablas como estructura de datos, estableciendo un conjunto de relaciones tabulares que resultan tan importantes como los propios datos. Así, la base de datos del sistema estará integrada por un conjunto de tablas, las

cuales representan diversos grupos de entidades y que además se pueden relacionar entre sí mediante campos comunes que indican la correspondencia entre los elementos de las diferentes entidades.

7.3.2 Diseño de la base de datos

El diseño de la base de datos es una fase crucial, pues se define completamente la estructura del conjunto de datos. Se ha decidido dividir este proceso en dos etapas, necesariamente en el orden que se indican. Primeramente, se ha llevado a cabo el diseño conceptual, el cual consiste en especificar toda la información que contendrá la base de datos en relación a las entidades, atributos y relaciones. Este diseño ofrece una visión clara y organizada de la estructura teórica de la base de datos. Posteriormente, se ha desarrollado el diseño lógico, donde se especifica detalladamente la estructura de la base de datos en función al modelo de base de datos relacional.

7.3.2.1. Diseño del modelo conceptual de la base de datos

Con el objetivo de elaborar el modelo conceptual de la base de datos del sistema se ha utilizado el modelo conceptual de Entidad-Relación (E-R) (Chen, 1976), debido a que es uno de los más conocidos, sencillos y útiles. Se basa en las entidades, las cuales tienen atributos y presentan entre sí relaciones de dependencia o correspondencia. Este proceso se ha llevado a cabo mediante el programa *Open ModelSphere* (3.2.2 Build 971 N), una herramienta CASE (*Computer Aided Software Engineering*) que facilita y automatiza las fases de diseño conceptual, diseño lógico e implementación de la estructura de la base de datos.

Así pues, se ha llevado a cabo la elaboración del modelo Entidad-Relación completo del sistema propuesto, priorizando tanto las entidades como las relaciones y poniendo especial énfasis en la cardinalidad y dependencia o correspondencia de estas últimas. Más aún, y con tal de proporcionar una visión clara de la estructura del conjunto de datos, se ha diferenciado cada tipología de entidad usuario, ruta, formulario y preguntas - mediante un color diferente. El diseño del modelo conceptual de la base de datos se contempla en la figura 3.

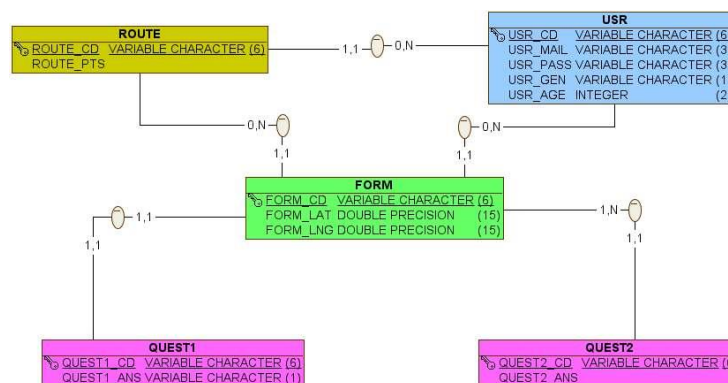


Figura 3. Esquema del modelo conceptual de la base de datos del sistema. Fuente: elaboración propia.

7.3.2.2. Diseño del modelo lógico de la base de datos

El objetivo establecido en esta fase fue definir la estructura de la base de datos del sistema según el modelo relacional de la base de datos, utilizando como base el diagrama conceptual E-R elaborado en el apartado anterior. Esto se basa en el hecho de que la elaboración del modelo conceptual mediante el modelo Entidad-Relación permite su traducción al modelo lógico, no de manera inmediata pero sí de manera sencilla. Este hecho es una de las principales razones que motivó la elaboración del modelo conceptual mediante el modelo E-R, demostrando

que es una herramienta potente para el diseño lógico de bases de datos. De nuevo toda esta fase se ha realizado mediante el software *Open ModelSphere*.

La construcción del modelo lógico de la base de datos se dividió en dos fases. En primer lugar (1), se llevó a cabo la completa especificación e introducción al modelo conceptual de todos los atributos, claves primarias y claves foráneas de las entidades del sistema, añadiendo también sus nombres físicos. Una vez introducidos todos los atributos y claves (2), se llevó a cabo la traducción del modelo conceptual al modelo lógico mediante el programa indicado. El diseño del modelo lógico de la base de datos se contempla en la figura 4.

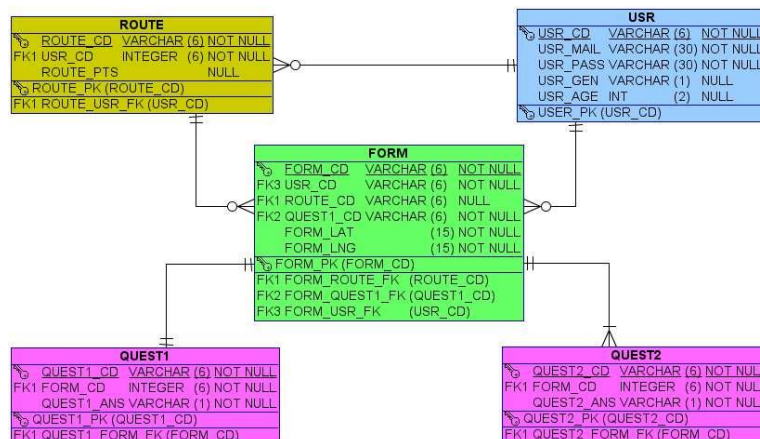


Figura 4. Esquema del modelo lógico de la base de datos del sistema. Fuente: elaboración propia.

8 IMPLEMENTACIÓN

8.1 Propósito del apartado

El objetivo del presente apartado es describir, en medida de lo posible, el proceso de programación e implementación del diseño funcional descrito a lo largo de todo el documento. De esta manera, se recoge desde una descripción de las diferentes tecnologías empleadas hasta la programación de la totalidad del sistema, pasando por la implementación y puesta en marcha de la base de datos e indicando diversos aspectos importantes relacionados con las especificaciones técnicas y métodos empleados.

8.2 Tecnologías empleadas

8.2.1 Desarrollo web

El desarrollo web es el término que define el proceso de construir y mantener sitios web, logrando una experiencia de usuario agradable en tanto que se consigue un funcionamiento rápido y un buen desempeño en términos de eficiencia. Para conseguirlo, se hace uso de tecnologías de software de lado del cliente – o *front-end* – y tecnologías de lado del servidor – o *back-end* –, involucrando una combinación de procesos de base de datos con el uso de un navegador web, con el objetivo de realizar tareas o mostrar información.

En lo que concierne al presente proyecto, se ha llevado a cabo tanto el desarrollo *front-end* como el desarrollo *back-end* del sistema, si bien resulta importante explicar antes en qué consiste cada término. Por un lado (1), el *front-end* se compone de aquellas tecnologías que se procesan en el lado del cliente y que por lo tanto están enfocadas en el usuario. Así, son las encargadas de ofrecer una buena experiencia al usuario, tanto en términos de diseño como en términos de interactividad. Por el otro lado (2), el *back-end* se integra por aquellas tecnologías

que se procesan en el lado del servidor, *detrás* del cliente. Así, se encarga de las consultas o peticiones a la base de datos, conexión al servidor o toma, procesado y retorno de datos al usuario. En la figura 5 - ver siguiente página - se observa un esquema de la descrita composición y estructura de un sitio web.

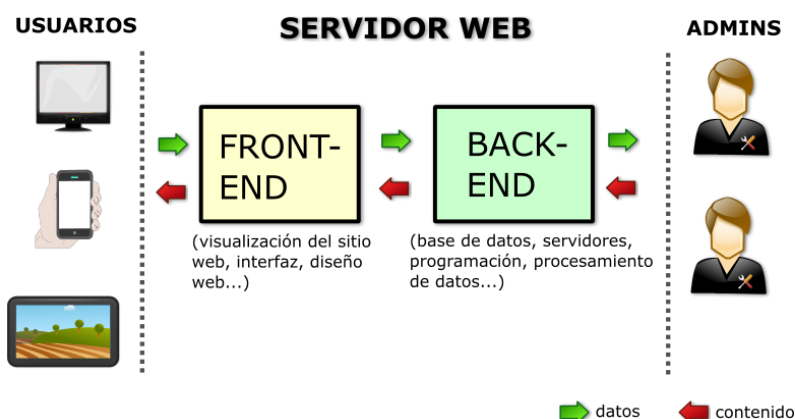


Figura 5. Esquema de la estructura de un sitio web, teniendo en cuenta tanto la vertiente front-end como back-end.
Fuente: Alegsa.com

8.2.1.1. Desarrollo web de lado de cliente

El conjunto de tecnologías y lenguajes de programación empleados en cuanto a *front-end* se contemplan a continuación.

- **HTML** (*HyperText Markup Language*): para definir el orden y contenido de las vistas.
- **CSS** (*Cascading Style Sheets*): para establecer el diseño visual de las vistas, con el objetivo de desarrollar una aplicación visualmente atractiva y mejorar así la experiencia de usuario.
- **JavaScript**: para mejorar la interfaz de usuario y especialmente para dotar de interactividad y funcionalidad tanto al cliente móvil como al cliente de escritorio, a través del procesamiento de actividades complejas.

Si bien el eje básico del desarrollo web de lado de cliente que se ha empleado se compone de los tres lenguajes descritos, se ha hecho uso de otra técnica de desarrollo denominada AJAX - *Asynchronous JavaScript And XML* -. Se trata de una tecnología asíncrona que permite desarrollar aplicaciones interactivas al lograr que el lado de cliente sea capaz de actualizarse continuamente sin la necesidad de refrescar la página completa. Así, se ha hecho uso de esta tecnología con el objetivo de crear unas aplicaciones más rápidas y con mayor capacidad de respuesta a las acciones de los usuarios.

8.2.1.2. Desarrollo web de lado de servidor

En lo relativo a los lenguajes y tecnologías empleados en el lado de servidor se ha utilizado básicamente PHP - *Hypertext Preprocessor* -. Se ha decidido emplear este lenguaje puesto que es un lenguaje muy popular y ampliamente utilizado en el *back-end*, por lo cual dispone de una extensísima comunidad de usuarios e incontables documentos de ayuda.

Más aún, y con el objetivo tanto de instalar PHP como de construir un servidor de datos, se ha utilizado el paquete de software XAMPP. Este paquete consiste en un único programa que permite descargar e instalar, de manera cómoda y simultánea, el servidor web Apache, por un lado, y el intérprete de scripts de PHP, entre otros componentes que no se han utilizado en el proyecto. Así, este paquete ha resultado de vital importancia puesto que todo el proceso de

desarrollo, implementación y pruebas del sistema se ha visto soportado por la existencia de un servidor como es Apache para permitir la comunicación entre la base de datos y los clientes.

Por último, y en lo relativo al sistema de gestión de base de datos, se ha decidido utilizar PostgreSQL en conjunción a PgAdmin 4, una aplicación gráfica que permite acceder a todas las funcionalidades de la base de datos.

8.2.2 Frameworks y bibliotecas

La principal biblioteca que se ha utilizado para desarrollar el sistema es la biblioteca multiplataforma jQuery, cuyo uso muestra numerosas ventajas. Por un lado (1), facilita enormemente el uso de la técnica AJAX ver 8.2.1.1 -, por el otro lado (2), simplifica en gran manera la manipulación DOM, facilitando la interacción con los documentos HTML, y por último (3) facilita también el desarrollo de animaciones e interacciones.

Por otro lado, con el objetivo de automatizar la adaptación de las vistas al formato de los dispositivos táctiles, se ha utilizado jQuery Mobile, un framework ligero fuertemente inspirado en jQuery que está especialmente diseñado para el desarrollo de aplicaciones web para móviles, en pos que pretende conseguir una misma sensación de navegación por parte del usuario final en la mayoría de dispositivos móviles.

Por último, merece mención el uso de Leaflet, una biblioteca de JavaScript destinada a facilitar el desarrollo de mapas en entornos web cuyo uso ha permitido la implementación de mapas tanto en el cliente web como en el cliente de escritorio del sistema. En apartados posteriores se describirán diversos paquetes externos que también merecen mención.

8.2.3 Entorno y plataforma de desarrollo

Dada la necesidad de poder adaptar el desarrollo web llevado a cabo al ámbito de los dispositivos móviles se ha utilizado la plataforma Adobe Phone Gap, un paquete de librerías que permite empaquetar aplicaciones web y ejecutarlas como si fueran aplicaciones nativas. La principal ventaja de esto es que permite desarrollar una aplicación multiplataforma, escribiendo el código una única vez y modificándolo solo ligeramente o incluso sin hacerlo para adaptarlo a los diferentes sistemas operativos móviles. El principal inconveniente, no obstante, es que no es tan eficiente en términos de uso de los recursos y herramientas del dispositivo móvil.

Por su lado, Adobe Phone Gap no es más que una distribución de Apache Cordova, un entorno de desarrollo móvil donde una de sus funcionalidades más atractivas es la de permitir la instalación de plugins, paquetes externos de código que ofrecen funcionalidades extras a la aplicación. Estos plugins han sido indispensables para la consecución de los objetivos establecidos, en pos de dotar al sistema de las funciones necesarias para llevar a cabo los procesos propuestos.

Por último, merece la pena señalar que se han utilizado dos editores de código distintos para llevar a cabo todo el proceso de implementación. Por un lado se ha utilizado mayoritariamente Visual Studio Code, un editor de código muy potente altamente personalizable y con una gran variedad de utilidades que facilitan el desarrollo de software. Por el otro lado, y en menor medida, se ha utilizado Notepad++, un editor de código muy simple, ligero y portable que soporta un sinnúmero de formatos y lenguajes de programación.

8.3 Estructura de directorios

8.3.1 Cliente móvil

La estructura de directorios de la aplicación móvil viene definida por la propia plataforma Phone Gap una vez se crea un proyecto. Sin embargo, si bien la estructura se ha mantenido prácticamente inalterada, se han añadido diversos módulos y directorios con tal de poder desarrollar el sistema correctamente. La estructura superior se recoge en la fig. 6 - ver siguiente página -, en la cual se observan un total de 5 directorios, de entre los cuales cabe destacar

platforms, *plugins* y *www*, si bien los otros dos directorios son resultado del proceso de empaquetamiento con *Node.js* y Phone Gap y no son importantes en lo que concierne a la descripción. En lo relativo a los directorios señalados, su uso es básicamente el siguiente:

- **Platforms:** almacena un directorio por cada plataforma para la cual se compilará la aplicación (Android, iOS, WP, etc).
- **Plugins:** almacena los plugins instalados.
- **www:** contiene la aplicación en su estructura web básica.

Es evidente que, de entre los directorios descritos, el directorio *www* resulta indispensable, puesto que es donde se almacena la estructura de las aplicaciones web desarrolladas y donde se encuentran los archivos a editar y elaborar. Así, se puede bajar un nivel dentro de dicho directorio, cuya estructura se recoge tamgién en la fig. 6 - ver abajo -.

De esta manera, y tal y como queda en evidencia, dicho directorio alberga a su vez un total de 10 subdirectorios, además de los 12 archivos HTML que conforman el armazón de la aplicación web, asignándose básicamente un archivo a cada *vista* de la aplicación a excepción de uno de ellos, que conforma el menú desplegable de la aplicación móvil. Teniendo esto en cuenta, la finalidad de los directorios es esencialmente el siguiente:

- **Bower_components:** recoge todo los archivos necesarios para la implementación del tema NativeDroid².
- **CSS:** recoge el archivo CSS del proyecto que añade o modifica estilos a los ya predeterminados por jQuery Mobile.
- **Data:** recoge el archivo de tipo *geojson* que almacena los polígonos de interés para el proyecto, como son plazas, parques y playas.
- **Img:** recoge las imágenes que se muestran en la aplicación.
- **jQuery:** recoge los archivos necesarios para implementar tanto jQuery como jQuery Mobile.
- **Js:** recoge el archivo JS en el que se define todas las funcionalidades y proceso que debe llevar a cabo la aplicación.
- **Leaflet:** recoge los archivos necesarios para implementar la biblioteca de creación de mapas interactivos Leaflet.
- **Leaflet_plugins:** recoge los plugins de Leaflet instalados.
- **Res y Spec:** se trata de carpetas creadas en el momento de crear el proyecto con Phone Gap. Contienen archivos necesarios para el correcto funcionamiento del sistema, además de los iconos de la aplicación dentro de Android.

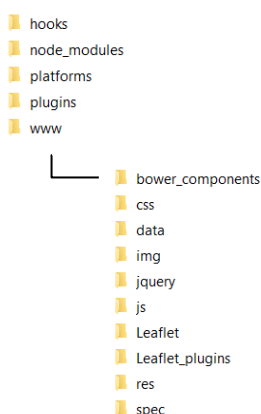


Figura 6. Estructura de directorios de la aplicación móvil

8.3.2 Cliente de escritorio

La estructura de directorios de la aplicación web de escritorio es prácticamente idéntica a la de la aplicación móvil, con la salvedad que es una versión más simple debido a que no ha sido creada con Phone Gap y que no está destinada a entornos móviles. Su estructura se recoge en la figura 7 - ver abajo - y la descripción de las funciones de cada directorio es idéntica a la realizada en el apartado anterior. Incluye un único archivo HTML denominado *Visor.html* que compone la aplicación de *back-office* desarrollada a efecto de control y explotación de los datos.

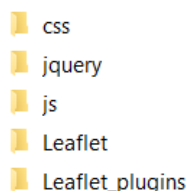


Figura 7. Estructura de directorios de la aplicación web

8.4 Implementación de la base de datos

Debido al empleo del programa *Open ModelSphere* para cubrir la etapa de diseño de la base de datos, el mismo programario permite automatizar la generación e implementación del esquema de la base de datos a partir del modelo lógico elaborado previamente. Así, el proceso llevado a cabo para implementar la base de datos en el entorno PostgreSQL queda contemplado en las siguientes líneas.

- i. **Generación del script SQL.** En este paso del proceso se lleva a cabo la conversión del modelo lógico relacional genérico, creado por defecto por el programa utilizado, al modelo lógico específico del SGBD de PostgreSQL. Posteriormente se procede a la conversión del modelo lógico específico a un *script* de SQL, el cual contiene las órdenes necesarias tanto para crear el conjunto de tablas que conformarán la base de datos como para establecer las relaciones que deberán existir entre ellas.
- ii. **Depuración y edición del script SQL.** En este momento se procede, esencialmente, a eliminar aquellos bloques de código residuales del proceso de conversión de modelo lógico a SQL y que resultan ser inservibles o incluso peligrosos, al poder inducir a errores en el momento de ejecutar el *script*.
- iii. **Construcción de las tablas de la base de datos en el entorno PostgreSQL.** En el paso final del proceso se da la conexión al servidor de la base de datos a través del programa web de administración de PostgreSQL, pgAdmin 4, para finalmente ejecutar el *script* de SQL creado y depurado previamente, implementando así el esquema de la base de datos diseñada dentro del entorno del SGBD.

8.5 Especificaciones empleadas

8.5.1 Formato de almacenamiento de datos

En lo relativo al almacenamiento de datos, ya sea tanto de manera local, en los clientes del sistema, como de manera remota, en la base de datos del mismo, se ha hecho uso de dos tipos de archivos muy parecidos el uno del otro, como son *json* y *geojson*. Resulta importante hacer una breve descripción de ambos formatos para ayudar a comprender el por qué de su uso.

Por su lado, el formato *json* (*JavaScript Object Notation*) es un formato ligero de intercambio de datos que permite intercambiar los mismos de una forma sencilla pero sobretudo

legible y comprensible. Se trata de un formato ampliamente utilizado en el entorno web a la hora de intercambiar información.

Por el otro lado, el formato *geojson* es un formato basado en *json* diseñado para representar elementos geográficos sencillos, junto con sus atributos no espaciales. Su gramática está basada en el estándar *Well-known text* - o WKT - del *Open Geospatial Consortium* o simplemente OGC y puede representar una geometría, un fenómeno o una colección de fenómenos. Se trata de un formato ampliamente utilizado en aplicaciones de cartografía en entornos web al permitir el intercambio de datos de manera rápida y comprensible, disponiendo por un lado de la geometría y por el otro lado de los atributos información alfanumérica asociados a esta información geográfica. La figura 8 - ver abajo - muestra un ejemplo de la gramática de ambas notaciones.

```
{ "name": "John", "age": 30, "car": null }

{
  "type": "Feature",
  "geometry": {
    "type": "Point",
    "coordinates": [125.6, 10.1]
  },
  "properties": {
    "name": "Dinagat Islands"
  }
}
```

Figura 8. Ejemplo de objetos con notación *json* (arriba) y notación *geojson* (abajo). Fuentes: *Geojson.org*, *W3C*.

De esta manera, lo interesante de estos formatos en lo que respecta al proyecto es que permiten, por un lado (1), obtener datos directamente del servidor y, por el otro lado (2), guardar esta información para generar mapas en un entorno web. Así, la aplicación específica de cada formato en el contexto del presente proyecto se describe a continuación.

- En lo que concierne a la notación *json*, esta se ha utilizado básicamente con dos propósitos. Por un lado (1) se emplea en los procesos de envío, almacenamiento y recuperación de las coordenadas de las rutas registradas en o desde la base de datos. Así, las coordenadas almacenadas se introducen en un objeto de tipo *json*, el cual se convierte a texto en el momento de enviarlo y almacenarlo en la base de datos. Posteriormente, en el momento de recuperarlo desde el servidor, el mismo objeto se *parsea* para transformarlo de texto a *json* y poder utilizarlo. Por el otro lado (2) se hace uso de esta notación para almacenar localmente diferentes parámetros necesarios para los diversos procesos del sistema. Entre estas variables encontramos, por ejemplo las propias coordenadas del usuario o el correo electrónico que ha facilitado en el momento de iniciar sesión.
- En lo relativo a la notación *geojson*, se ha utilizado este formato con tres motivos básicos. Por una parte (1), se ha utilizado este formato para almacenar localmente el total de polígonos que conforman las áreas de interés como son playas, parques o plazas, y donde en el momento de entrar en uno de estas áreas debe saltar un formulario. El motivo de uso del formato *geojson* con este propósito es que permite comprobar de manera rápida y eficiente si la posición del usuario, convertida en punto, se encuentra dentro de cualquiera de los polígonos del archivo. Por la otra parte (2), se ha hecho uso de este formato con el simple objetivo de mostrar en un mapa web las rutas y formularios registrados. Esto se debe, básicamente, a la discordancia en lo que concierne al uso de la latitud o

coordenada Y - y la longitud o la coordenada X - en el momento de definir unas coordenadas. Así, mientras que en España en general se definen las coordenadas como Longitud-Latitud, o X-Y, Leaflet define las coordenadas con el orden inverso. Por ende, si se almacena la información geográfica de la manera convencional, en el momento de querer mostrarla en un mapa web mediante Leaflet esta información se mostrará en otro lugar y no en el ámbito de la ciudad de Barcelona. En cambio, si se hace uso de un *geojson* en el momento de querer pintar la información geográfica en un mapa web, esta discordancia se corrige automáticamente, permitiendo visualizar correctamente la información relativa a rutas y formularios. Así, el proceso que se lleva a cabo para pintar rutas y registros en los mapas web del sistema consiste básicamente en recuperar las coordenadas del servidor, en formato *json*, e introducirlas en un *geojson*, consiguiendo así visualizar la información geográfica en los visores.

Por último (3), se ha decidido que el formato *geojson* sea el formato de descarga de datos por parte de los investigadores del GEMOTT, visto que es un formato de fácil elaboración directamente desde el lado de cliente y que además está soportado en un ampliamente utilizado software SIG como es QGIS. Así, y también a través de la biblioteca *FileSaver.js*, se ha conseguido implementar la descarga de información de la base de datos directamente desde el cliente de escritorio, de manera que los investigadores pueden disponer en cualquier momento de un archivo *geojson* con información de las rutas y otro con información de los formularios respondidos, siendo dichos archivos visualizables y explotables desde QGIS.

La totalidad de la descripción realizada queda resumida en la figura 9 - ver abajo -.

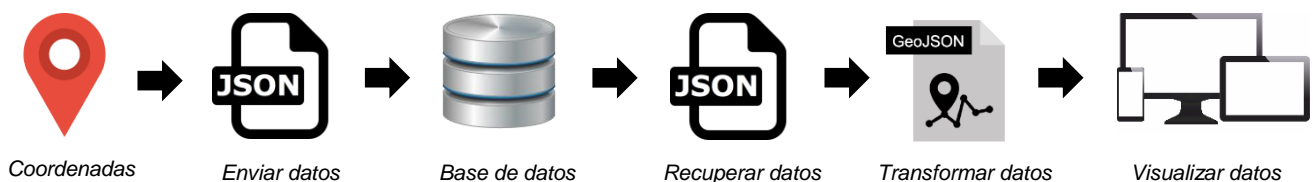


Figura 9. Esquema resumen del proceso llevado a cabo para enviar, almacenar, recuperar y visualizar la información geográfica del sistema. Fuentes: elaboración propia.

8.5.2 Análisis geoespacial

Con el objetivo de llevar a cabo distintos procesos, como pueden ser desplegar un formulario en el momento en que el usuario entra en una área de interés o calcular la distancia total recorrida en una ruta, se ha decidido utilizar *Turf.js*, una biblioteca de Javascript destinada al análisis geoespacial en entornos web. Así, y como se ha introducido, se ha empleado dicha biblioteca con dos objetivos principales, descritos a continuación.

- Por un lado (1), *Turf.js* permite realizar distintos tipos de análisis topológico, donde el más importante en lo que concierne al sistema es el de comprobar si un polígono contiene un punto. Lo que se lleva a cabo, básicamente, es convertir de manera constante cada nueva posición del usuario en un punto y comprobar si dicho punto se encuentra contenido en cualquiera de los polígonos recogidos en el *geojson* de datos especificado en el apartado anterior. Así, y en función del resultado del análisis, se da una respuesta u otra.
- Por el otro lado (2), la biblioteca permite calcular, de manera sencilla, la distancia total que se recorre en una ruta, al calcular la distancia existente entre coordenadas. Esto resulta se emplea básicamente como medio informativo para los usuarios e información adicional para el equipo del GEMOTT.

8.6 Precisión de posicionamiento

La precisión de los datos relacionados con la posición del usuario depende enteramente tanto de la ubicación y entorno del usuario como, más importante, del propio dispositivo móvil y de su capacidad de recibir información de las constelaciones satelitales disponibles en un momento y lugar determinados. De esta manera, y teniendo en cuenta la tipología del receptor GNSS y que es poco probable que el usuario disponga de sistemas adicionales de posicionamiento, como podría ser una red Wifi, se espera que la precisión de posicionamiento del usuario oscile entre los 5 y los 20 metros en condiciones normales.

9 RESULTADOS

9.1 Vistas de usuario

9.1.1 Cliente móvil

La aplicación móvil desarrollada cuenta con un total de 11 vistas distintas. De esta manera, se cuentan un total de 6 interfaces de más respecto al prototipado inicial llevado a cabo ver apartado 7.2 y donde además el diseño final, si bien mantiene la esencia la propuesta inicial, se ha modificado notablemente. De esta manera, la totalidad de las vistas del cliente móvil se contemplan entre las figuras 10 y 20 - ver siguientes páginas -, mayoritariamente ordenadas según su orden de aparición conforme el usuario hace uso de la aplicación.

Así, en las figuras 10 y 11 se muestra, respectivamente, la vista de registro de usuario y la vista de inicio de sesión. La función de dichas vistas es muy simple, donde en la primera el usuario debe introducir algunos datos personales como género y edad y las credenciales de inicio de sesión que quiere utilizar en la aplicación, como son un correo electrónico y una contraseña. En la segunda vista, de inicio de sesión, el usuario debe introducir dichas credenciales para poder acceder a la aplicación.

The image displays two mobile application screens side-by-side. Both screens have a white background and a header with the GEMOTT logo (a stylized orange square with a white figure) and the text 'GEMOTT GRUP D'ESTUDIS DE MOBILITAT, TRANSPORT I TERRITORI'.
The left screen, titled 'Registre', contains three input fields: 'Correu electrònic' (Email), 'Contrasenya' (Password), and 'Edat' (Age). Below these is a 'Gènere' (Gender) section with two buttons: 'HOME' and 'DONA'. At the bottom, there is a link 'Al registrar-te, acceptes els termes de privacitat' and a 'Ja estàs registrat?' link, followed by a large teal button labeled 'REGISTRAR-ME'.
The right screen, titled 'Iniciar sessió', contains two input fields: 'Correu electrònic' (Email) and 'Contrasenya' (Password). Below these is a link 'No estàs registrat?'. A large teal button labeled 'INICIAR SESSIÓ' is positioned below the input fields. At the bottom, the UAB logo (orange 'U' and 'A' with a blue 'B') is displayed above the text 'Universitat Autònoma de Barcelona'.

Figura 10 (izquierda). Vista de registro de usuarios. Figura 11 (derecha). Vista de inicio de sesión.

Por su lado, en las figuras 12 y 13 se muestra el menú principal de la aplicación, tanto de manera completa como detrás del menú desplegable, donde este también es observable desde la vista de estadísticas. De esta manera, la función de ambos menús es ayudar al usuario a dirigirse a las vistas o funciones que desee, como por ejemplo puede ser comenzar un registro de ruta, ver sus estadísticas, acceder a una vista de los términos de uso y privacidad figura 14 -, en la cual se expone el destino de los datos obtenidos y los derechos de los usuarios en cuanto a su información, o acceder a una vista de ayuda de la aplicación figura 15 en la cual se muestra una pequeña guía de uso de la aplicación.

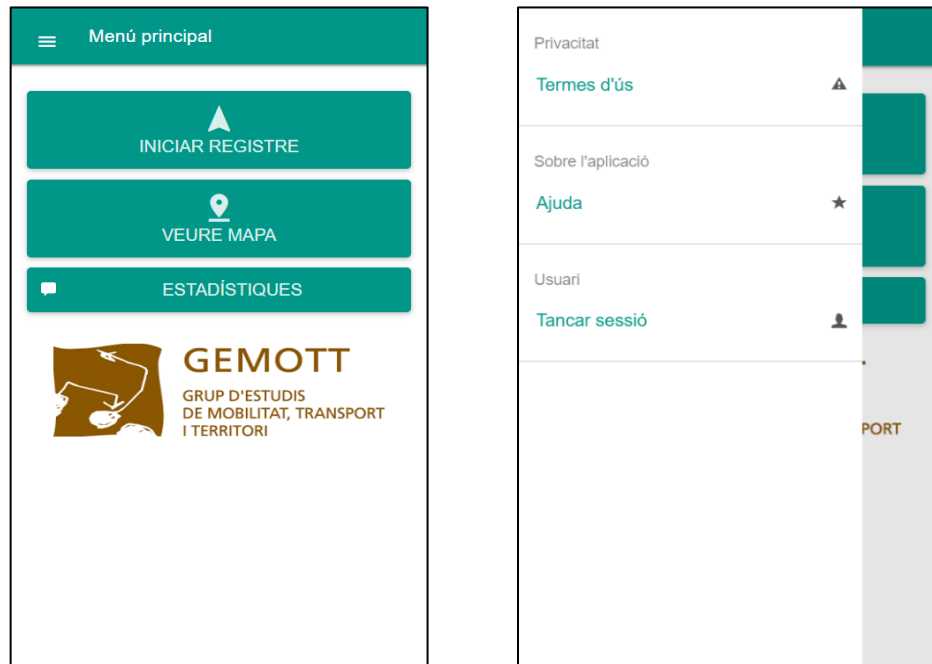


Figura 12 (izquierda). Menú principal de la aplicación. Figura 13 (derecha). Menu principal de la aplicación con el menú lateral desplegado.



Figura 14 (izquierda). Vista de términos de uso y privacidad de la aplicación. Figura 15 (derecha). Vista de ayuda de la aplicación

Por el otro lado, en las figuras 16 y 17 se muestra, respectivamente, la vista de estadísticas del usuario y la vista de registros realizados por el usuario en un mapa de Barcelona. Nótese que en este caso ambas vistas no muestran información alguna pues solo se contemplan a modo de ejemplo. Así, en la primera interfaz el usuario puede ver diversas estadísticas relacionadas con el uso de la aplicación, como pueden ser los formularios respondidos, las rutas registradas o la distancia total recorrida en el total de rutas. En la segunda interfaz, por su lado, se deberían ver sobre el mapa tanto las rutas como la ubicación de los formularios respondidos. Para ver un ejemplo de esta tipología de resultados, ver el apartado 9.3.

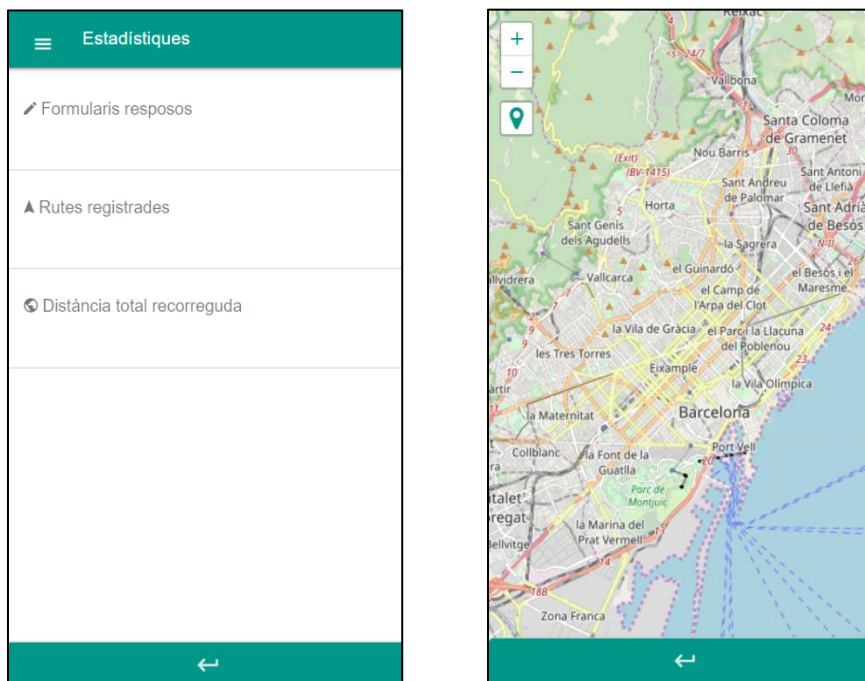


Figura 16 (izquierda). Vista de estadísticas del usuario. Figura 17 (derecha). Vista del mapa de Barcelona.

Más aún, las figuras 18, 19 y 20 se relacionan ya con el uso del acelerómetro para detectar parades por parte del usuario. De esta manera, la figura 16 es la vista correspondiente a un dialogo de decisión que salta cuando la aplicación detecta que el dispositivo se ha parado durante 30 segundos. Así, en esta vista el usuario puede decidir si quiere responder un formulario, relativo a cómo le hace sentir el espacio donde está tal y como se muestra en la figura 17 -, dar la ruta actual por finalizada y enviarla a la base de datos o por el contrario continuar caminando. Siguiendo este contexto, y posteriormente a responder la primera pregunta del formulario, aparece la vista correspondiente a la segunda pregunta del formulario figura 18 , relativa al por qué le hace sentir el espacio donde està como ha indicado en la pregunta anterior.

T'has aturat?

Sembla ser que t'has aturat. Vols respondre un formulari, acabar la ruta o continuar caminant?

+

RESPONDRE FORMULARI

ACABAR RUTA

CONTINUAR CAMINANT

Com et fa sentir aquest lloc?

☒
MOLT BÉ

☐
BÉ

☐
INDIFERENT

☐
MALAMENT

☐
MOLT MALAMENT

Figura 18 (izquierda). Vista de dialogo de decisión cuando el usuario se para. Figura 19 (derecha). Primera pregunta del formulario que deben responder los usuarios.

Per què?

☐
ARBRES I PLANTES

☐
COMERÇOS

☐
BANCS / MOBILIARI

☐
ESTAT DE LES VORERES

☐
GENT

☐
TRÀNSIT

☐
SOROLL

☐
OLOR

☐
IL·LUMINACIÓ

☐
NETEJA

☐
ALTRES

Figura 20. Segunda pregunta del formulario que deben responder los usuarios.

9.1.2 Cliente de escritorio

La aplicación web de escritorio cuenta con una única vista, tal y como se denota en la figura 21. Dicha vista consiste en un visor en el cual se muestra un mapa de la ciudad de Barcelona, también sobre una base de *Open Street Maps*. Nótese, como en los ejemplos anteriores, que no se muestra ningún tipo de información pues es solamente una vista de ejemplo. De esta manera, la gran diferencia respecto a la vista del mapa del cliente de escritorio es la presencia del botón de descarga de la esquina superior izquierda, diferenciado con una esfera del mundo. La función de dicho botón es iniciar la descarga de información desde la base de datos en formato *geojson* para que pueda ser explotado por parte del equipo investigador.

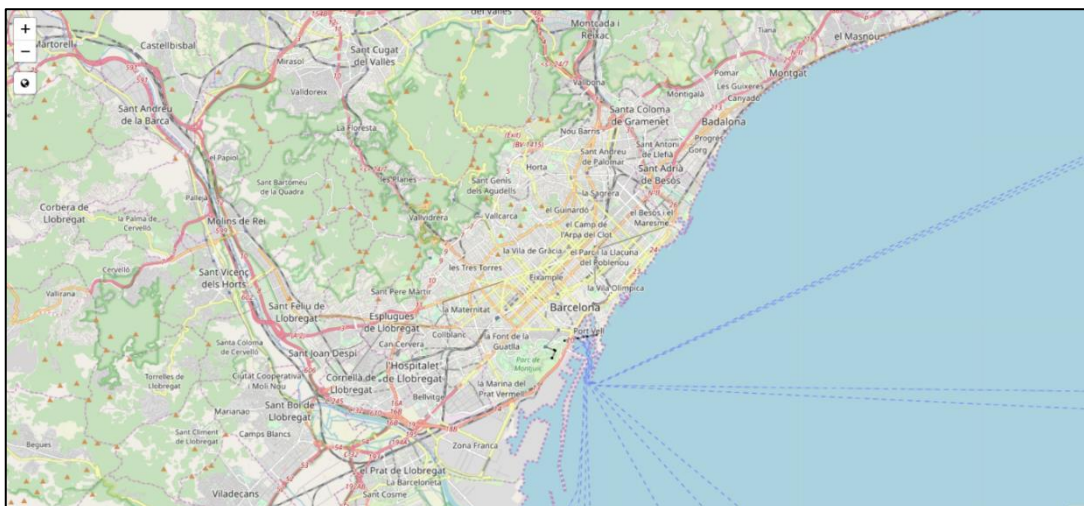


Figura 21. Visor de back-office del grupo de investigadores del GEMOTT, centrado en la Ciudad de Barcelona.

9.2 Obtención de datos

el menú principal ver figura 12 -. Así, y una vez se haya desplazado a la vista del mapa de Barcelona, el usuario podrá centrar el mapa en su ubicación, si bien hasta que no se desplace la aplicación no comenzará a registrar su posición ver figura 22, abajo -. Así, y una vez empiece a desplazarse, el usuario podrá ver a la vez cómo su ruta se *pinta* en pantalla y los formularios que ha respondido ver figura 23 -. Una vez finalizada la ruta, ya sea porque el usuario así lo desea o porque se ha parado definitivamente, la misma quedará registrada en la base de datos si hay conexión a internet. Los formularios, por su parte, se han ido registrando y almacenando en la base de datos conforme se han respondido.

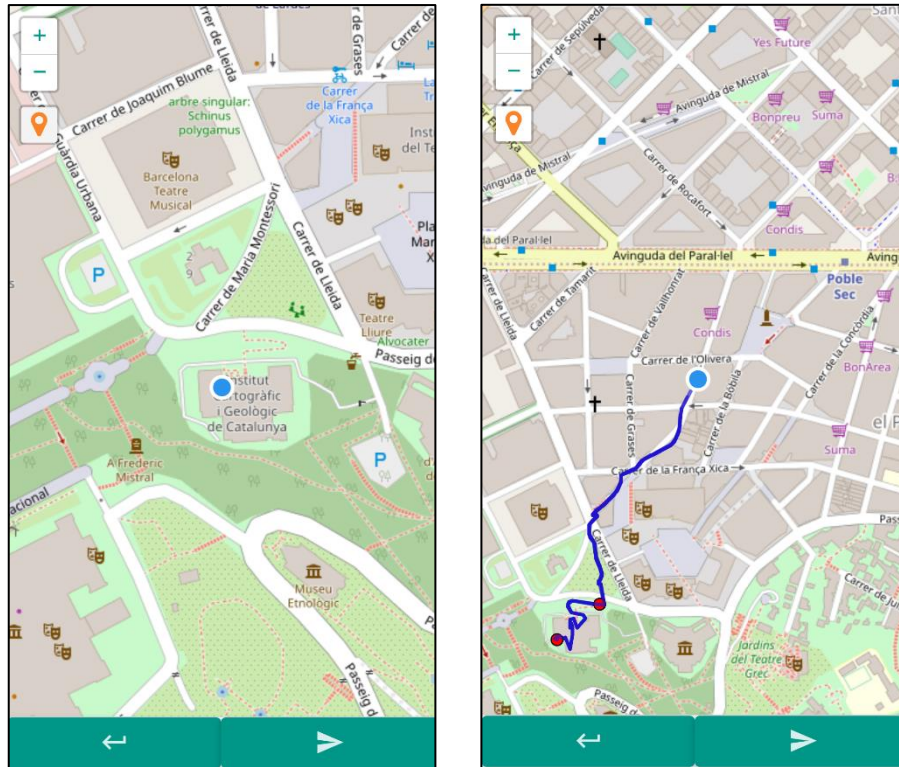


Figura 22 (izquierda). Vista de registro de ruta del usuario, centrada en su posición. Figura 23 (derecha). Vista de registro de ruta del usuario, con su ruta recorrida y formularios respondidos.

9.3 Retorno de datos

El retorno de datos se puede dar por cuatro vías diferentes, donde dos son para los usuarios y dos para el equipo de investigadores del GEMOTT. Así, y en lo que respecta al retorno de datos al equipo del GEMOTT, este se da principalmente a través de la aplicación web de escritorio, donde pueden ver todas las rutas y formularios registrados en la base de datos ver figura 24 -. De esta manera, y a través de este visor, pueden hacer *click* tanto sobre las rutas como sobre los formularios para ver información sobre ellos a través de un *popup*, como es la distancia total de la ruta ver figura 25 - o las respuestas dadas en los formularios ver figura 26 -.

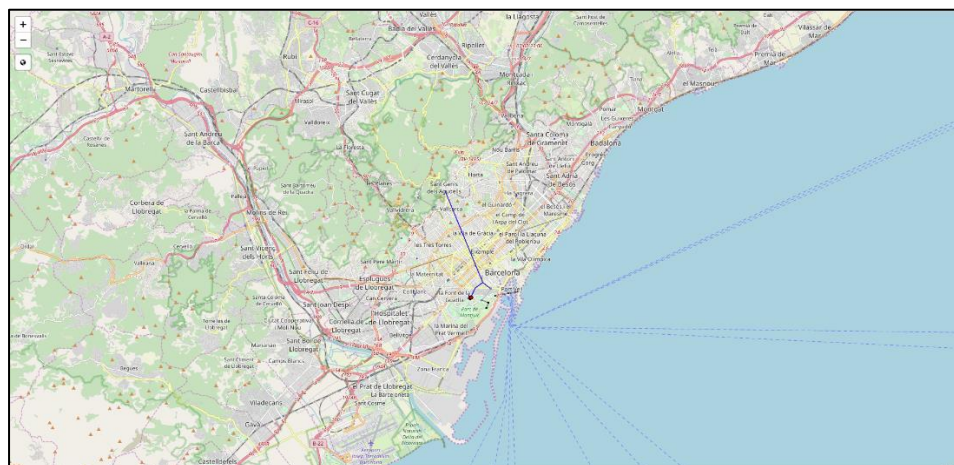


Figura 24. Visor de back-office del grupo de investigadores del GEMOTT, con las rutas y formularios registrados en la base de datos sobre el mapa de Barcelona.

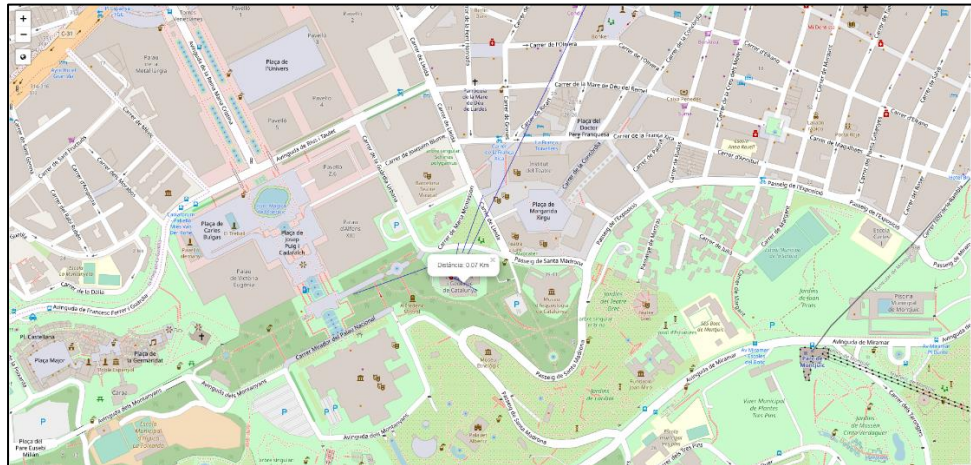


Figura 25. Visor de back-office del grupo de investigadores del GEMOTT, con un popup que muestra la distancia recorrida en la ruta elegida.



Figura 26. Visor de back-office del grupo de investigadores del GEMOTT, con un popup que muestra las respuestas dadas en el formulario elegido.

Más aún, el equipo del GEMOTT puede utilizar el botón destinado a la descarga de datos para descargar todo el contenido de la base de datos del sistema ver figura 27 -. Esta descarga se da en formato *geojson*, dando lugar a un archivo con información de las rutas y otro archivo con información de los formularios. Dichos archivos se pueden abrir directamente en un cliente como QGIS o transformar a otros formatos, en función de las necesidades.

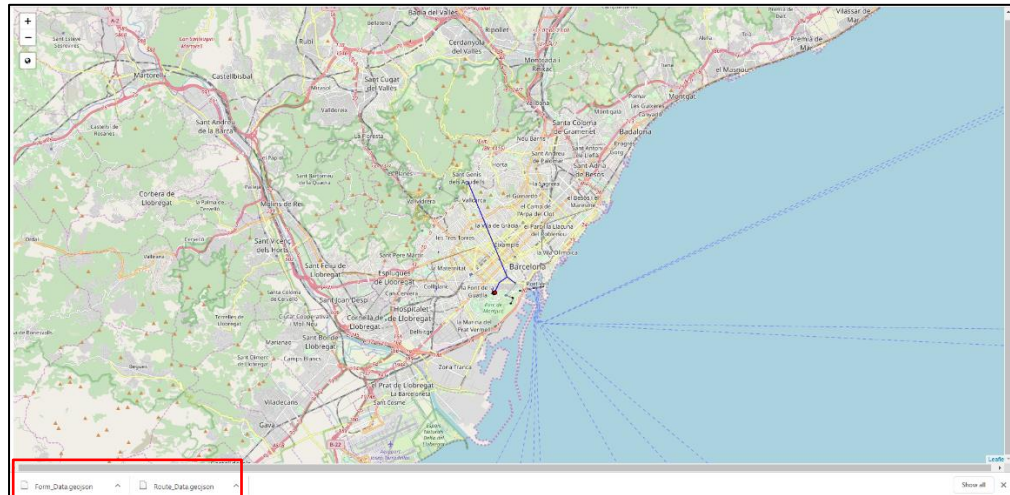


Figura 27. Visor de back-office del grupo de investigadores del GEMOTT, donde se puede observar la descarga de datos a través de dos ficheros en formato geojson

Por el otro lado, y en lo que respecta al retorno a los usuarios, estos pueden ver tanto las rutas realizadas georreferenciadas como la posición de los formularios que han respondido, además de la distancia recorrida por cada ruta y las respuestas que han dado a los formularios, a través de la vista del mapa ver figura 28 -. La información para rutas y formularios, que se pinta en pantalla a través de un *popup*, se muestra de manera idéntica a como se ha mostrado para la aplicación de escritorio de los administradores. Más aún, los usuario tambien pueden ver toda esta información recogida en la vista de estadísticas, donde pueden ver el número de rutas que han enviado, el número de formularios respondidos y la distancia total que han recorrido ver figura 29 -.

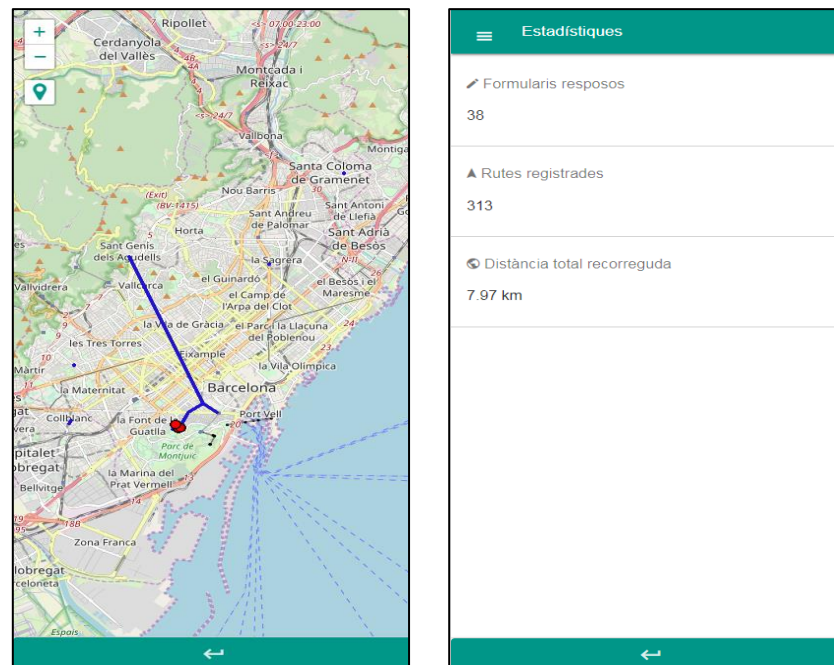


Figura 28 (izquierda). Vista del mapa de Barcelona con las rutas y formularios registrados por el usuario. Figura 29 (derecha). Estadísticas del usuario.

10 CONCLUSIONES

El objetivo principal del proyecto era desarrollar e implementar un sistema completo de orientada a servicios, llevando a cabo desde el diseño y construcción de la base de datos hasta el desarrollo de dos clientes, uno móvil y otro de escritorio, pasando por la implementación del servidor y la total funcionalidad del sistema. De esta manera, y ya en el punto final del proyecto, se ha conseguido cumplir el objetivo establecido en un principio, pues se cuenta con un sistema que representa, por sí mismo, una herramienta sólida de obtención de datos que se espera pueda funcionar de manera perlongada y ofrezca la máxima ayuda al equipo investigador del GEMOTT.

Siguiendo en esta línea, ha sido el propio cliente final el que ha puesto a prueba el sistema desarrollado, tanto en su versión de cliente móvil como de cliente de escritorio. En general han quedado satisfechos con las funcionalidades del sistema, donde además su ayuda ha sido indispensable para asegurar el funcionamiento del sistema y detectar errores de funcionamiento, sobretodo relacionados con el diseño gráfico del cliente móvil o con el inicio de sesión en el mismo.

Por el otro lado, y haciendo autocrítica, si bien es cierto que se ha conseguido implementar el sistema propuesto y desarrollar las funcionalidades demandadas, también es cierto que existe un aspecto que debería ser objeto de mejora en proyectos futuros como es la seguridad del sistema. Aunque es cierto que requiere la autenticación del usuario, y según su perfil se le muestran las rutas que ha recorrido y los formularios que ha respondido, el proceso de autenticación en el que el usuario envía su correo electrónico y contraseña se realiza sin ningún tipo de encriptación, por lo que cualquier ente malintencionado podría ser capaz de capturar estos datos.

Más aún, y en términos del marco del proyecto dentro del máster cursado, se puede afirmar que el presente trabajo ha servido como un proceso tanto de aprendizaje de nuevas prácticas como de repaso e implementación de los conocimientos adquiridos a lo largo de todo el curso, donde se ha dado un contacto inicial con lo más parecido al mundo laboral en el ámbito del desarrollo de software, concretamente en las geoaplicaciones, que ha consistido por sí mismo en una muy enriquecedora experiencia. Se puede concluir que se han alcanzado todos los objetivos propuestos, tanto en términos de desarrollo como en términos de aplicación de los conocimientos adquiridos.

11 BIBLIOGRAFÍA

11.1 Libros y artículos

- Barot, J. 2018. *Estudi dels entorns saludables per a la gent gran mitjançant tecnologia Smartphone*. Universitat Autònoma de Barcelona.
- Chen, P. (1976). *The Entity Relationship Model – Toward a Unified View of Data*. ACM Transactions on Database Systems. Vol. 1 (1).
- Codd, E. F., 1970. *A relational model of data for large shared data banks*. Communications of the ACM. 13(6):377 387.
- Gabino, G. *Qué son las metodologías ágiles y su aplicación en el mercado actual*. Capgemini. Universitat de València.
- Marquet, O., Alberico, C., Hipp, A. J., 2018. *Pokémon GO and physical activity among college students. A study using Ecological Momentary Assessment*. Computers in Human Behavior, Volume 81, Pages 215-222.
- Moskowitz, D. S., Young, S. N., 2006. *Ecological momentary assessment: what it is and why it is a method of the future in clinical psychopharmacology*. J Psychiatry Neurosci. 31(1): 13 20.
- Nixon, R., 2014. *Learning PHP, MySQL & JavaScript: With jQuery, CSS & HTML5 (Learning Php, Mysql, Javascript, Css & Html5)*. O'Reilly Media; Edición: 4.
- Sánchez, P. *Ingeniería de Software II. Tema 07: Gestión de riesgos en Proyectos Software*. Universidad de Cantabria.
- Trigas, M. *Metodología SCRUM. Desarrollo detallado de la fase de aprobación de un proyecto informático mediante el uso de metodologías ágiles*. Gestión de proyectos informáticos.
- Van de Ven, P Nelson, J., Rocha, A., 2017. *ULTEMAT: A mobile framework for smart ecological momentary assessments and interventions*. Internet Interventions. Volume 9, Pages 74-81.

11.2 Sitios web

- Alegsa <<http://www.alegsa.com.ar>>
- Arsys <<https://www.arsys.es>>
- Coding Infinite <<http://jezzsystemzone.blogspot.com>>
- DevCode <<https://devcode.la>>
- Draw <<https://draw.io>>

- FileSaver.js <<https://github.com/eligrey/FileSaver.js/>>
- GIS Stack Exchange <<https://gis.stackexchange.com>>
- Google Sites <<https://sites.google.com>>
- Grado y Máster Multimedia UOC <<http://multimedia.uoc.edu>>
- Heflo <<https://www.heflo.com>>
- Iwun UK <<https://www.iwun.uk/shmapped>>
- Jtech <<http://www.jtech.ua.es>>
- Leaflet <<https://leafletjs.com/>>
- Mapiness <<https://www.mapiness.org.uk>>
- Mockup Builder <<https://www.mockupbuilder.com>>
- Mozilla Developer <<https://developer.mozilla.org>>
- NativeDroid² <<https://github.com/wildhaber/nativeDroid2>>
- Platzi <<https://platzi.com>>
- Ser programador <<https://serprogramador.es>>
- Stack Overflow <<https://stackoverflow.com>>
- System Zone <<http://jezzsystemzone.blogspot.com>>
- Turf.js <<https://turfjs.org/>>
- Webucator <<https://www.webucator.com>>
- Wikipedia <<https://es.wikipedia.org>>

11.3 Aplicaciones móviles de referencia

- Bline <<https://bline.io>>.
- Mobile Market Monitor <<https://www.mobilemarketmonitor.com>>
- Mosquito Alert <<http://www.mosquitoalert.com/ca>>
- Paco App <<https://pacoapp.com>>
- Urban Mind <<https://www.urbanmind.info>>

ANEXOS

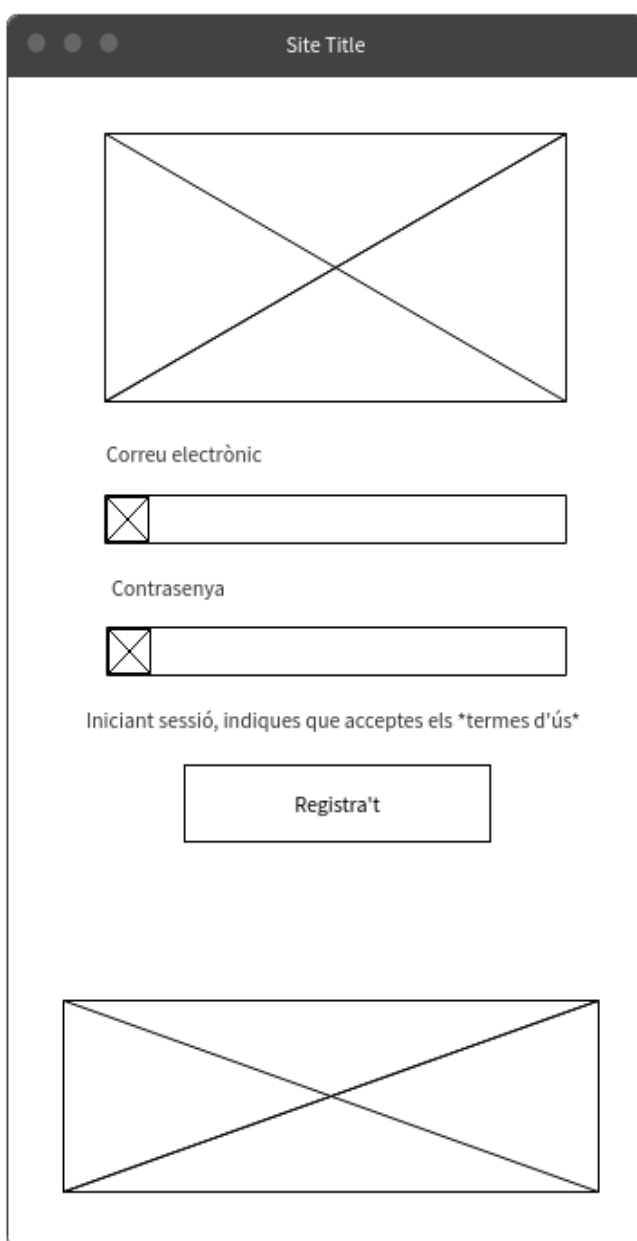
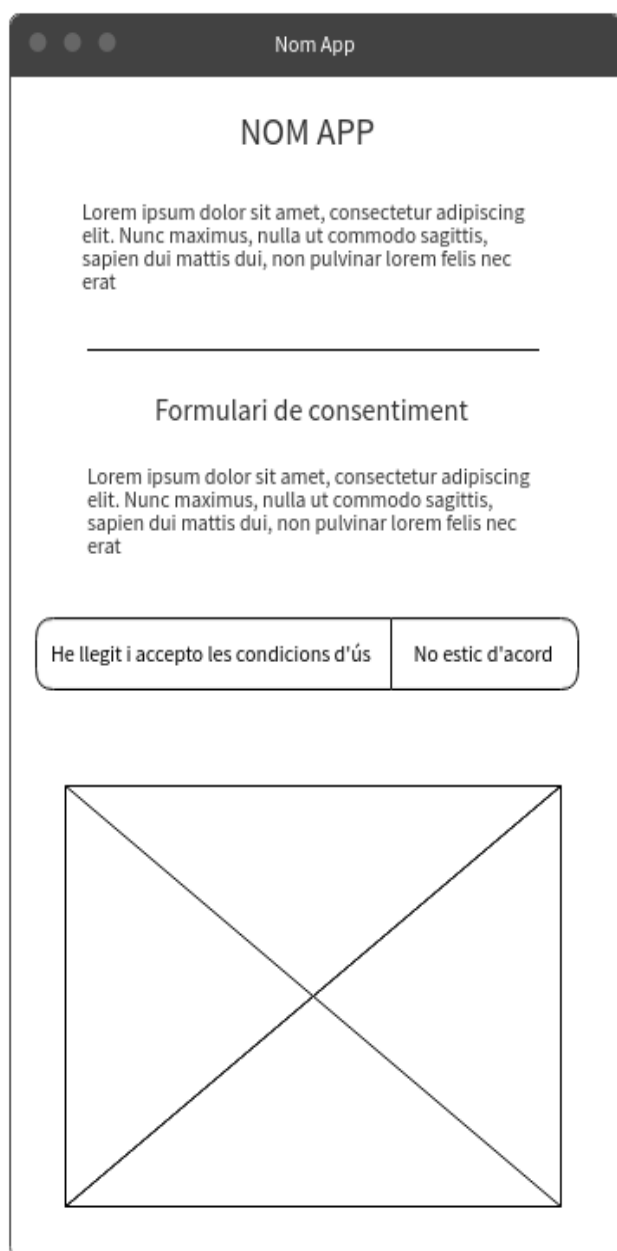


Figura 1 del anexo (izquierda). Wireframe de la vista de consentimiento del cliente móvil. Figura 2 del anexo (derecha). Wireframe de la vista de registro del cliente móvil.

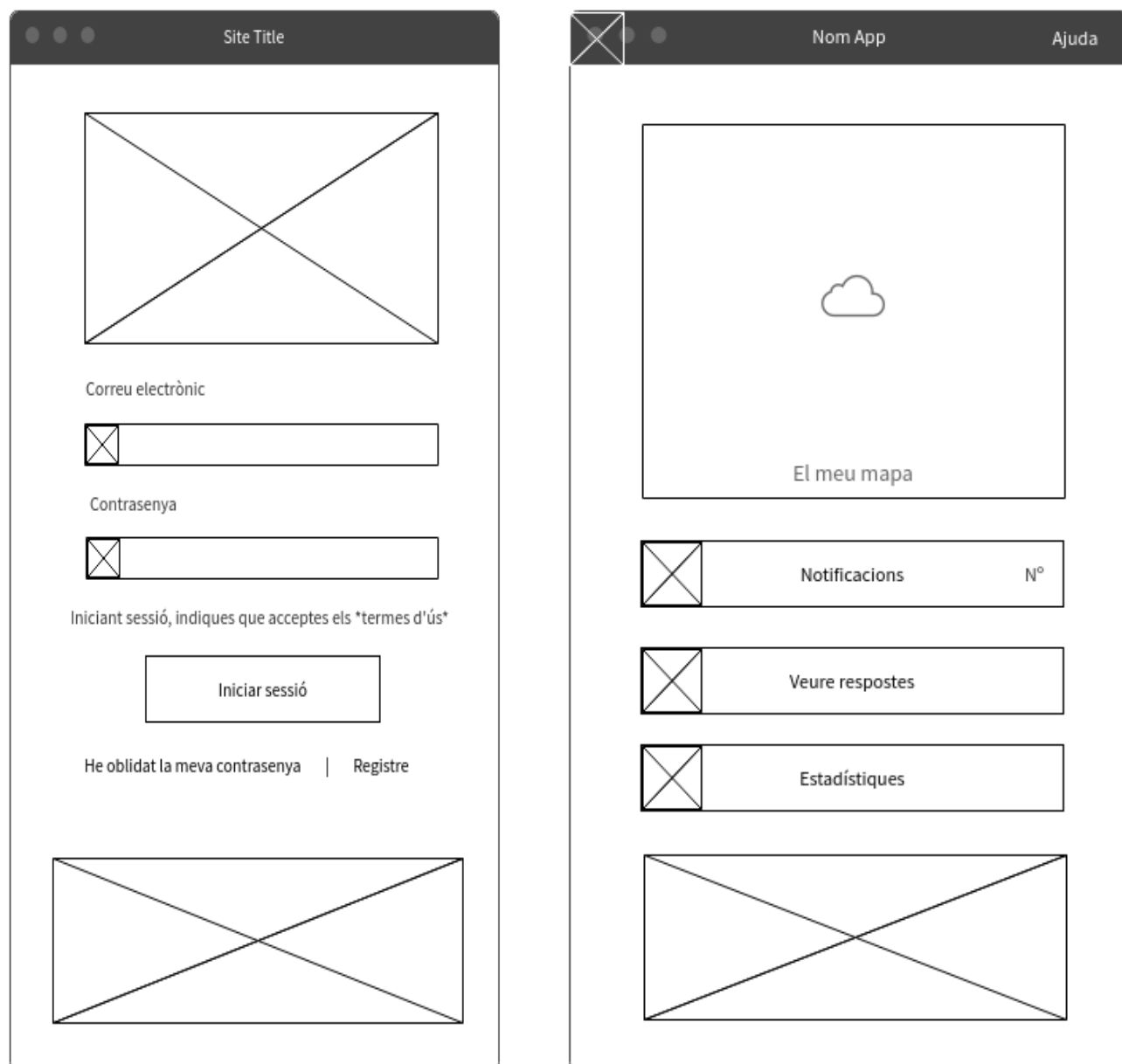


Figura 3 del anexo (izquierda). Wireframe de la vista de inicio de sesión del cliente móvil. Figura 4 del anexo (derecha). Wireframe de la vista del menú principal del cliente móvil.



Figura 5 del anexo. Wireframe de la vista de inicio de sesión del cliente móvil.

GLOSARIO DE ÍNDICES

Tablas

Grupo	Descripción	Funciones de uso
Desarrolladores	Los miembros del equipo de desarrollo son los encargados de desarrollar las aplicaciones siguiendo un correcto proceso de ingeniería del software. De entre las distintas fases que contempla el proyecto, la prueba del sistema constituye una parte crítica en pos de asegurar el correcto funcionamiento de este. Así, los desarrolladores también se convierten en usuarios, puesto que están obligados a hacer las pertinentes pruebas de funcionamiento.	Desarrollo y <i>test</i> del software.
Usuarios	Los usuarios o <i>peatones</i> son el público objetivo a los cuales está destinada la aplicación móvil. Deben ser capaces de hacer un uso simple de la misma, pudiendo tanto geolocalizar sus rutas a pie como resolver formularios. Disponen, además, de la posibilidad de visualizar las rutas que han recorrido a través un mapa dinámico.	Público objetivo de la aplicación móvil. Interacción con la lógica del software, en concreto con la geolocalización de sus rutas a pie y de la resolución de formularios.
Investigadores	Los investigadores o <i>administradores</i> son los encargados de gestionar y explotar toda la información y datos recogidos a través de la aplicación móvil, en relación tanto a las rutas como a las respuestas de los formularios. Son, además de los clientes finales y público objetivo de la totalidad del sistema, los usuarios de la aplicación web de <i>backoffice</i> . Por ello, se les puede considerar el grupo de usuarios principal. Más aún, los investigadores también se pueden convertir en usuarios en el caso que deseen participar en las pruebas de funcionamiento.	Público objetivo de la aplicación web y de la totalidad del sistema. Gestión y explotación de la base de datos. Interacción con la lógica del software, en concreto con la visualización de rutas geolocalizadas y las respuestas a los formularios.

Tabla 1. Definición de los tipos de usuarios del sistema desarrollado, juntamente a sus funciones de uso

ID	Actividad	Fecha de inicio	Fecha final	Predecesora
A	Especificar requisitos y evaluar proyecto	08/04/2019	30/04/2019	-
A-I	Elaboración del cronograma	08/04/2019	08/04/2019	-
A-II	Recolección de requisitos	09/04/2019	28/04/2019	A-I
A-III	Especificación preliminar del software	15/04/2019	28/04/2019	A-I
A-IV	Revisión de especificaciones	29/04/2019	29/04/2019	A-II, A-III
A-V	Incorporar mejoras en requisitos	29/04/2019	30/04/2019	A-IV
B	Análisis y diseño	01/05/2019	08/05/2019	A
B-I	Diseño de la base de datos	01/05/2019	03/05/2019	A-IV
B-II	Diseño de la interfaz de la aplicación móvil	01/05/2019	03/05/2019	A-IV
B-III	Diseño de las funcionalidades de las herramientas a implementar	03/05/2019	05/05/2019	B-I, B-II
B-IV	Desarrollo de prototipos	03/05/2019	05/05/2019	B-I, B-II
B-V	Revisión de diseños	06/05/2019	06/05/2019	B-III
B-VI	Incorporar mejoras en diseño	06/05/2019	08/05/2019	B-IV
C	Implementación	09/05/2019	17/06/2019	B
C-I	Preparación de la estructura de directorios	09/05/2019	10/05/2019	B-V
	Preparación del entorno de librerías			
	Codificación de la estructura HTML del total de vistas de la aplicación			
C-II	Implementación de la base de datos	09/05/2019	10/05/2019	B-V
C-III	Codificación del mapa	11/05/2019	14/05/2019	C-I, C-II
	Codificación de las herramientas de navegación			
	Codificación del control topológico			
C-IV	Revisión de implementaciones	15/05/2019	15/05/2019	C-III
C-V	Codificación de los formularios	16/05/2019	27/05/2019	C-IV
	Codificación del almacenamiento de la información			
C-VI	Codificación del pintado de datos en el mapa de la aplicación móvil	28/05/2019	02/06/2019	C-V
C-VII	Revisión de implementaciones	03/06/2019	03/06/2019	C-VI
C-VIII	Codificación del mapa de la aplicación web	03/06/2019	16/06/2019	C-VII
	Codificación del pintado de datos en el mapa de la aplicación web			
	Codificación de la descarga de datos a través de la aplicación web			
C-IX	Revisión de implementaciones	06/06/2019	06/06/2019	C-VI
C-X	Revisión de implementaciones	17/06/2019	17/06/2019	C-VIII
D	Pruebas unitarias y de integración	17/06/2019	09/07/2019	C
D-I	Estudiar módulos de código	17/06/2019	26/06/2019	C-VI
D-II	Probar módulos y sus componentes	20/06/2019	30/06/2019	D-I
D-III	Detectar fallos	01/07/2019	02/07/2019	D-II
D-IV	Corregir fallos	03/07/2019	07/07/2019	D-III
D-V	Volver a probar código corregido	08/07/2019	10/07/2019	D-IV
D-VI	Revisión del producto	11/07/2019	11/07/2019	D-V
E	Entrega	12/07/2019	12/07/2019	D
E-I	Entrega del producto	12/07/2019	12/07/2019	D-VI

Tabla 2. Planificación de las actividades a llevar a cabo en el proyecto

ID	Prioridad	Descripción
1	M	Se deben geolocalizar las rutas que recorren los usuarios
2	M	Deben aparecer formularios cuando el usuario haga una pausa o entre en una zonas de interés
3	M	Se debe enviar y almacenar la información en una base de datos de manera segura.
4	M	Se debe poder descargar la información almacenada en la base de datos
5	S	Se debe activar automáticamente el posicionamiento del usuario cuando este se comience a desplazar.
6	C	La aplicación se debe ejecutar en segundo plano.
7	C	El equipo de investigadores puede disponer de una aplicación de <i>back-office</i> consistente en un visor web de escritorio donde se pueda ver la información almacenada en la base de datos.

Tabla 3. Pila de producto del sistema desarrollado

Nombre	Cliente	Función
Locate	Ambos	Geolocalizar el usuario.
Turf.js	Ambos	Realizar análisis geoespaciales en la web. En el caso concreto del proyecto, detectar cuando el usuario ha entrado en un polígono de interés.
FileSaver	Escritorio	Permitir la creación y descarga de archivos desde el lado de cliente. En el caso del proyecto, facultar la descarga de datos en formato geojson.

Tabla 4. Definición y funciones de los paquetes externos empleado en el desarrollo del sistema.

Descripción del riesgo	Consecuencia	Probabilidad	Impacto	Mitigación	Afectación
Actualización de las bibliotecas externas utilizadas	Aplicación inoperativa	Baja	Alto	Utilización de bibliotecas de pago o últimas versiones estables	Coste
Ausencia de conexión a internet	Aplicación inoperativa	Media	Alto	Descarga y almacenamiento de datos del servidor en el dispositivo móvil	Entrega
Porcentaje de procesos erróneos mayor al 1% de los procesos realizados	Descontento del cliente y de los usuarios. Mal funcionamiento de las aplicaciones	Baja	Medio	Aumentar el número de las pruebas de calidad	Entrega / Reputación
Fallo de seguridad	Robo de datos de los usuarios. Descontento de los usuarios	Baja	Medio	Mantener el anonimato de los usuarios para no relacionar su información con ellos	Entrega
Falta de precisión de posicionamiento del dispositivo móvil de los usuarios	Información obtenida poco representativa	Media	Alto	Aumentar el número de muestras para asegurar la representatividad de los datos	Resultados

Tabla 5. Registro de riesgos y mitigaciones detectados del sistema desarrollado

Figuras

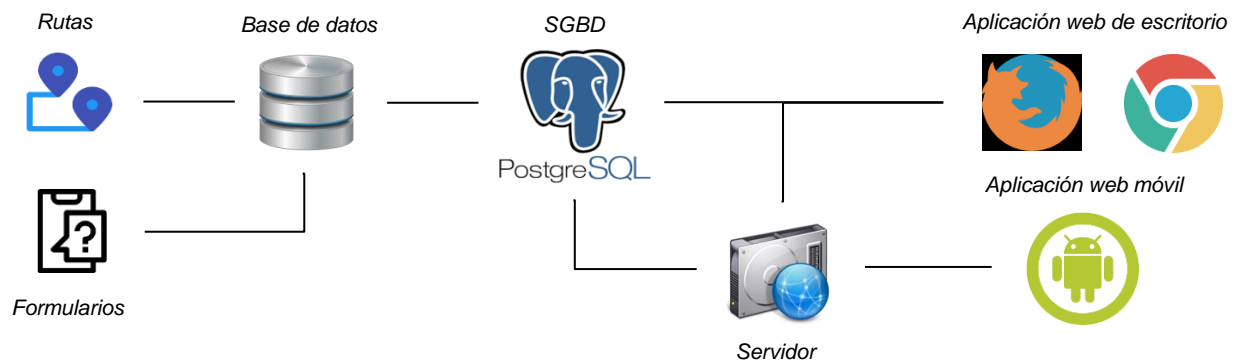


Figura 1. Esquema de la arquitectura del sistema desarrollado

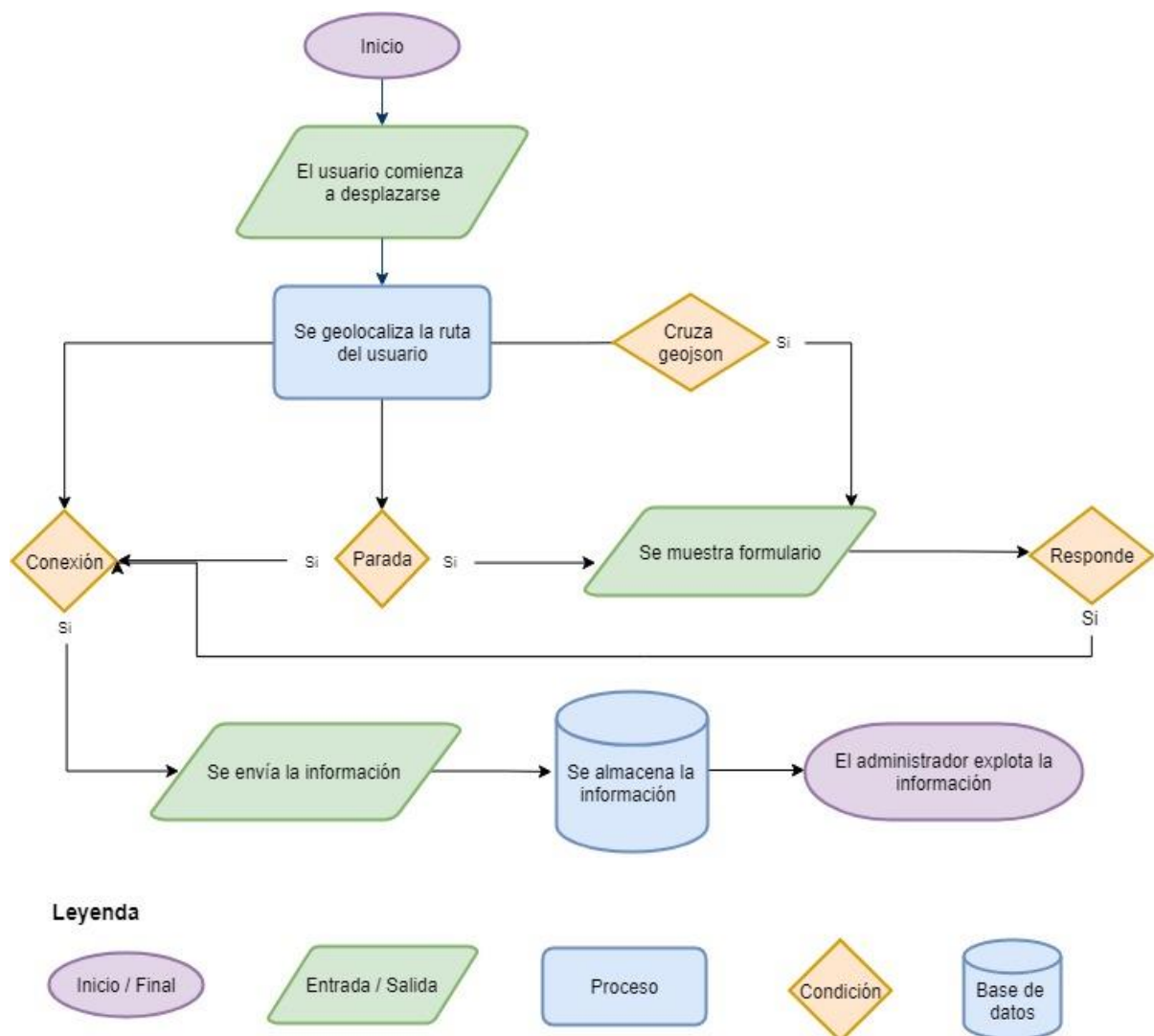


Figura 2. Diagrama de flujo del proceso principal de recogida y explotación de información del sistema desarrollado.
Fuente: elaboración propia.



Figura 3. Matriz DAFO para análisis estratégico diseñada para el proyecto desarrollado. Fuente: PlanillaExcel.com

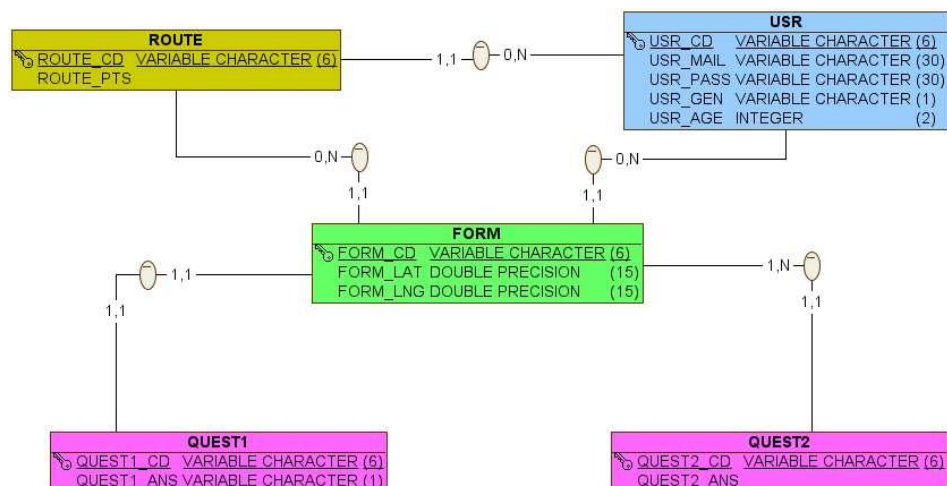


Figura 3. Esquema del modelo conceptual de la base de datos del sistema.

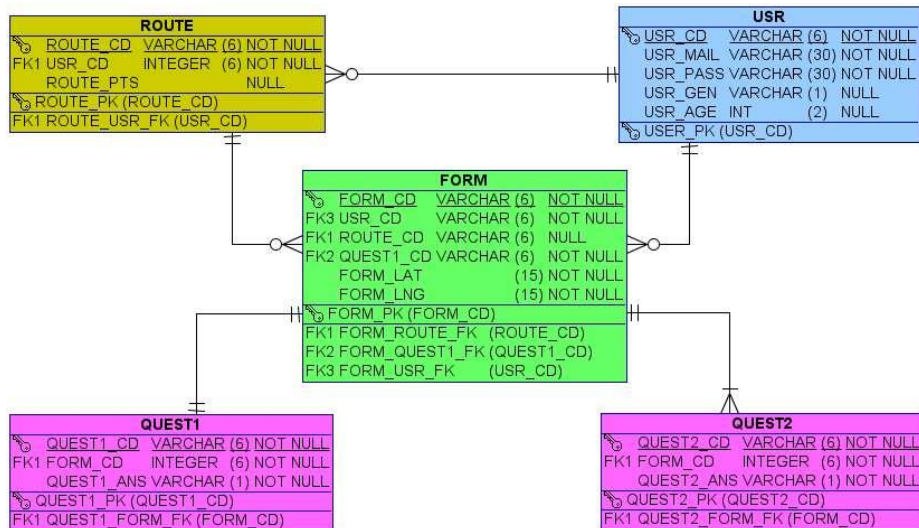


Figura 4. Esquema del modelo lógico de la base de datos del sistema

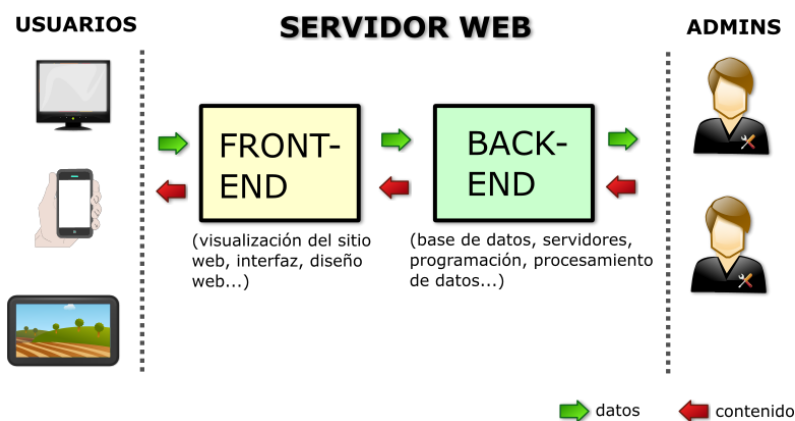


Figura 5. Esquema de la estructura de un sitio web, teniendo en cuenta tanto la vertiente front-end como back-end.
Fuente: Alegsa.com

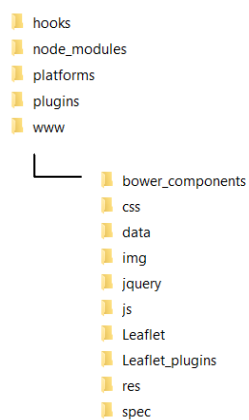


Figura 6. Estructura de directorios de la aplicación móvil

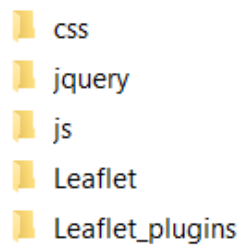


Figura 7. Estructura de directorios de la aplicación web

```

{ "name": "John", "age": 30, "car": null }

{
  "type": "Feature",
  "geometry": {
    "type": "Point",
    "coordinates": [125.6, 10.1]
  },
  "properties": {
    "name": "Dinagat Islands"
  }
}

```

Figura 8. Ejemplo de objetos con notación json (arriba) y notación geojson (abajo). Fuentes: Geojson.org, W3C.

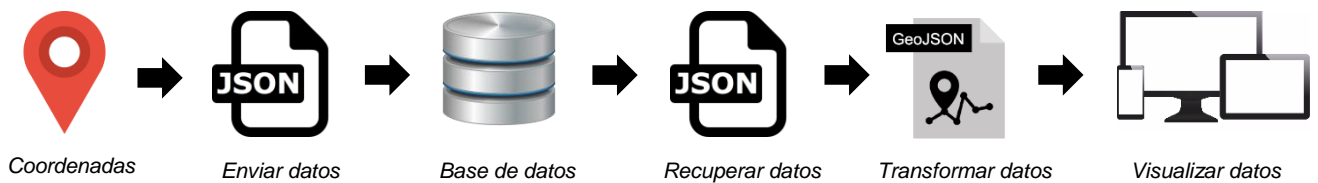



Figura 9. Esquema resumen del proceso llevado a cabo para enviar, almacenar, recuperar y visualizar la información geográfica del sistema. Fuentes: elaboración propia.



GEMOTT
GRUP D'ESTUDIS
DE MOBILITAT, TRANSPORT
I TERRITORI

Registre

Correu electrònic
Introduir correu electrònic...

Contrasenya
Introduir contrasenya...

Edat
Introduir edat...

Gènere
HOME DONA

Al registrar-te, acceptes els [termes de privacitat](#)

Ja estàs [regidrat?](#)

REGISTRAR-ME



GEMOTT
GRUP D'ESTUDIS
DE MOBILITAT, TRANSPORT
I TERRITORI

Iniciar sessió

Correu electrònic
Introduir correu electrònic...

Contrasenya
Introduir contrasenya...

No estàs [regidrat?](#)

INICIAR SESSIÓ

UAB
Universitat Autònoma
de Barcelona

Figura 10 (izquierda). Vista de registro de usuarios. Figura 11 (derecha). Vista de inicio de sesión.

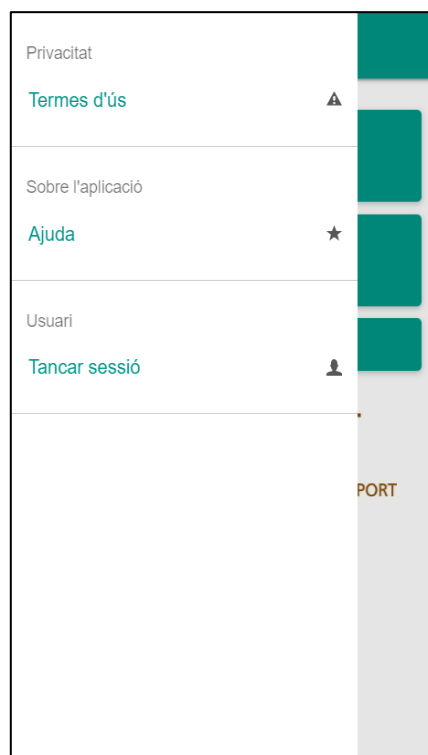
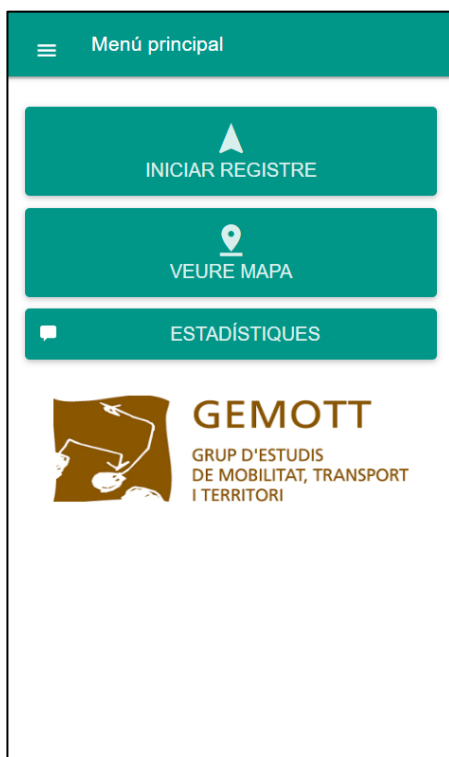


Figura 12 (izquierda). Menú principal de la aplicación. Figura 13 (derecha). Menu principal de la aplicación con el menú lateral desplegado.

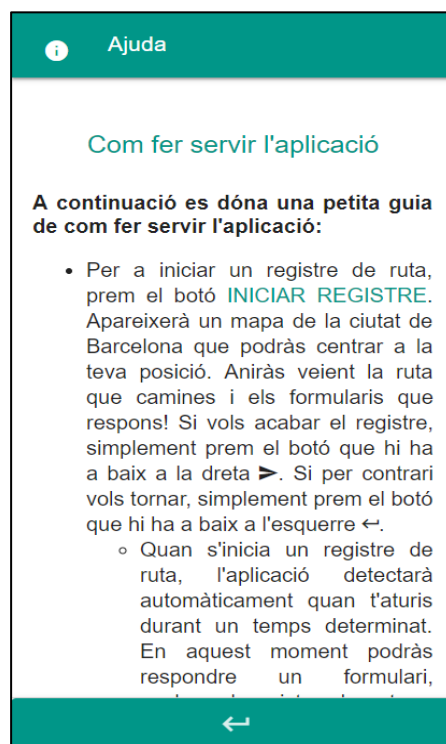
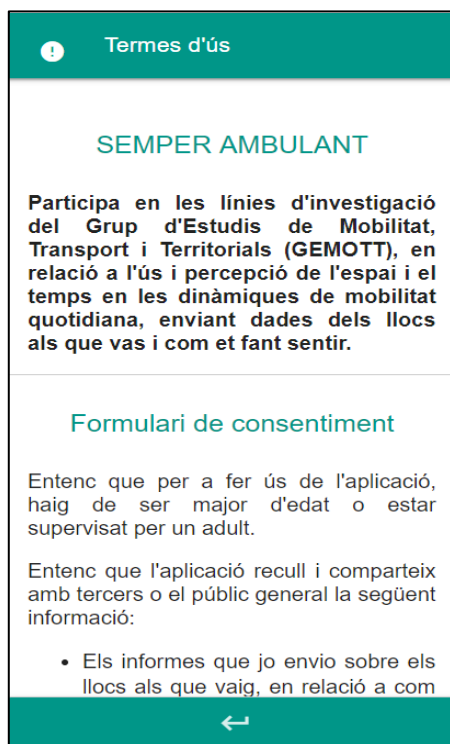


Figura 14 (izquierda). Vista de términos de uso y privacidad de la aplicación. Figura 15 (derecha). Vista de ayuda de la aplicación

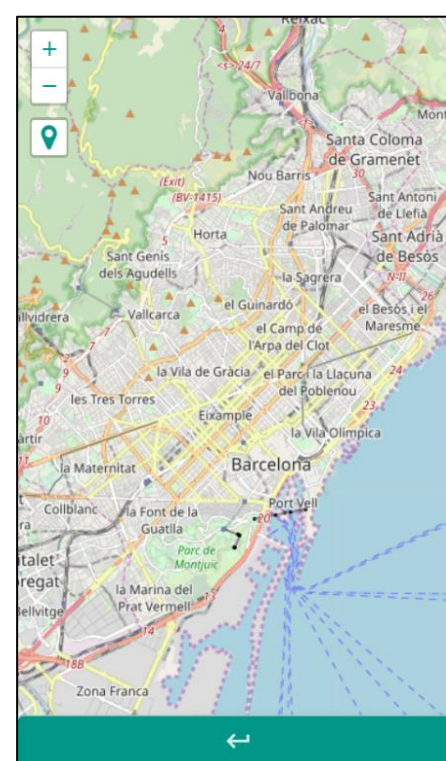




Figura 16 (izquierda). Vista de estadísticas del usuario. Figura 17 (derecha). Vista del mapa de Barcelona.



T'has aturat?


Sembla ser que t'has aturat. Vols respondre un formulari, acabar la ruta o continuar caminant?

+

RESPONDRE FORMULARI


ACABAR RUTA


CONTINUAR CAMINANT


Com et fa sentir aquest lloc?

☒
MOLT BÉ

☐
BÉ

☐
INDIFERENT

☐
MALAMENT

☐
MOLT MALAMENT







Figura 18 (izquierda). Vista de dialogo de decisión cuando el usuario se para. Figura 19 (derecha). Primera pregunta del formulario que deben responder los usuarios.


Per què?

☐
ARBRES I PLANTES

☐
COMERÇOS

☐
BANCS / MOBILIARI

☐
ESTAT DE LES VORERES

☐
GENT

☐
TRÀNSIT

☐
SOROLL

☐
OLOR

☐
IL·LUMINACIÓ

☐
NETEJA

☐
ALTRES




Figura 20. Segunda pregunta del formulario que deben responder los usuarios.

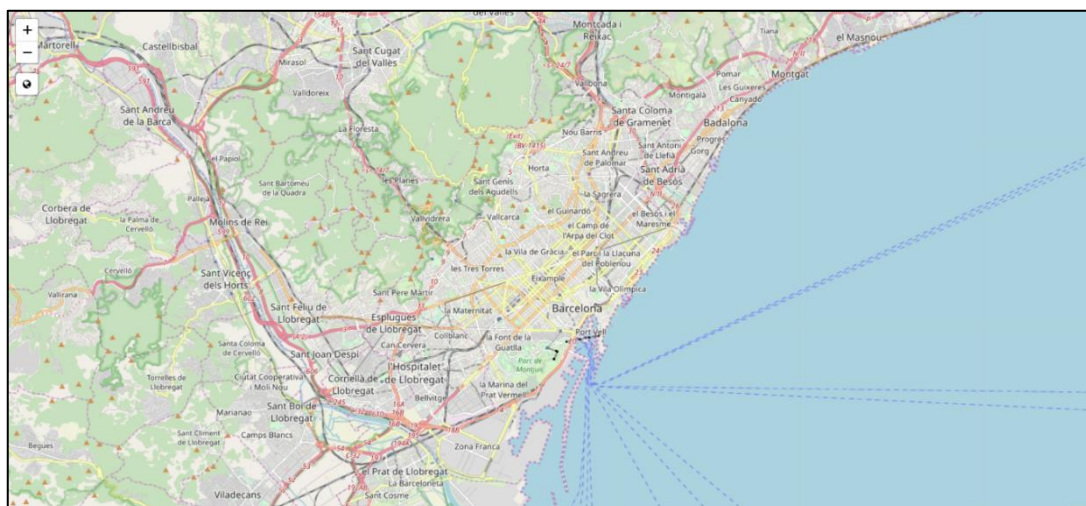


Figura 21. Visor de back-office del grupo de investigadores del GEMOTT, centrado en la Ciudad de Barcelona.

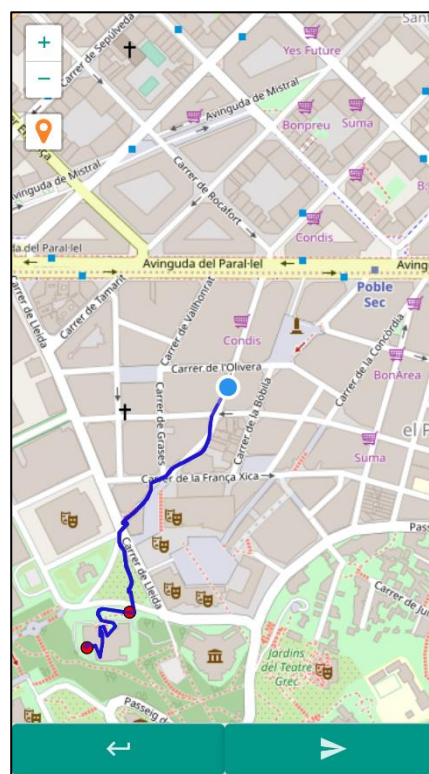
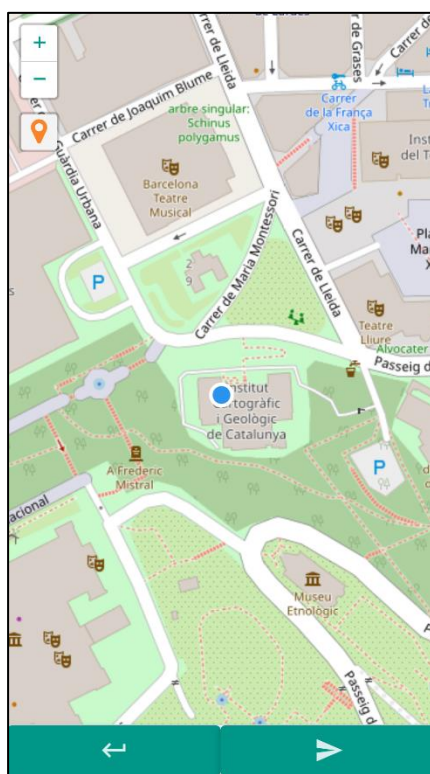


Figura 22 (izquierda). Vista de registro de ruta del usuario, centrada en su posición. Figura 23 (derecha). Vista de registro de ruta del usuario, con su ruta recorrida y formularios respondidos.

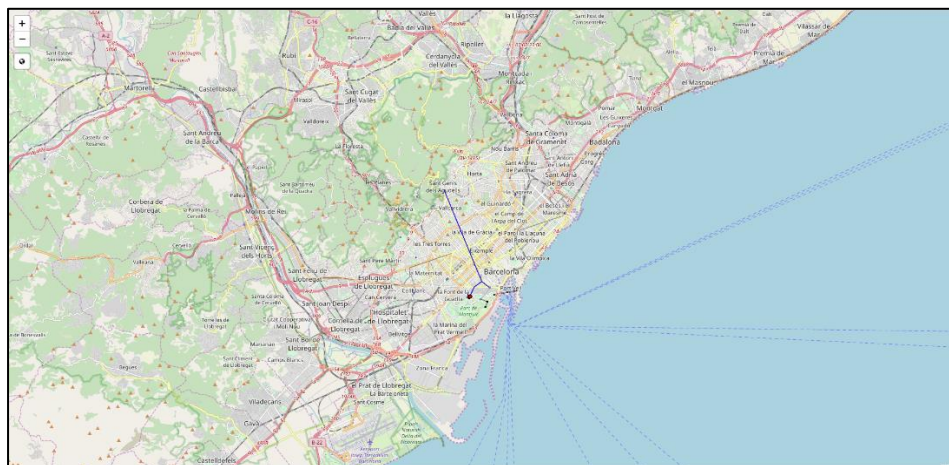


Figura 24. Visor de back-office del grupo de investigadores del GEMOTT, con las rutas y formularios registrados en la base de datos sobre el mapa de Barcelona.



Figura 25. Visor de back-office del grupo de investigadores del GEMOTT, con un popup que muestra la distancia recorrida en la ruta elegida.

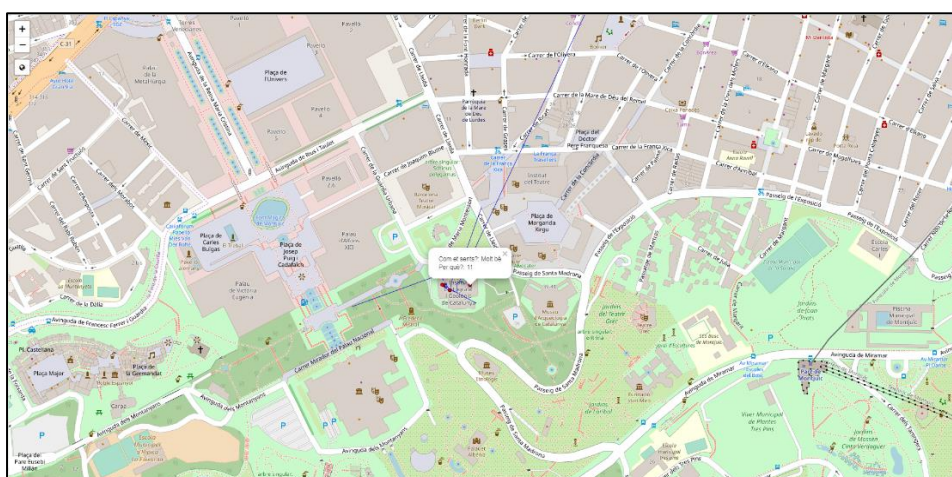


Figura 26. Visor de back-office del grupo de investigadores del GEMOTT, con un popup que muestra las respuestas dadas en el formulario elegido.

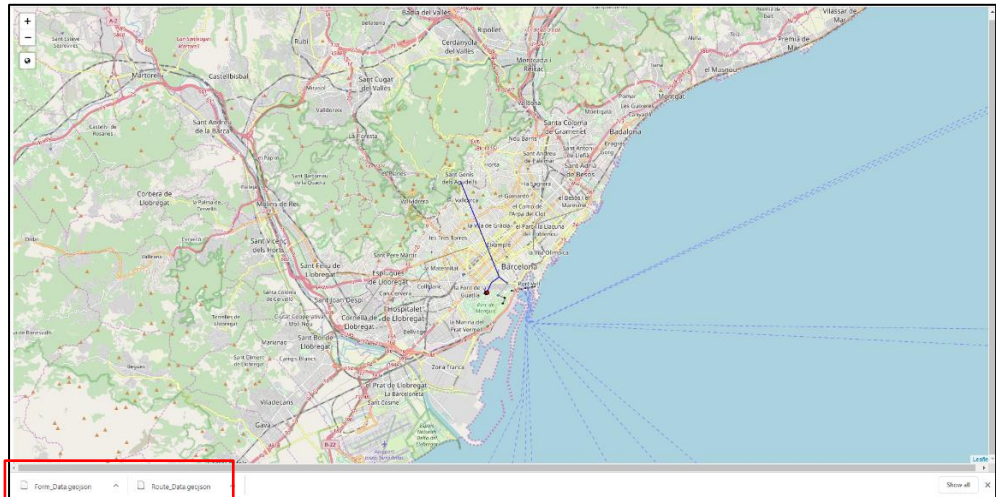


Figura 27. Visor de back-office del grupo de investigadores del GEMOTT, donde se puede observar la descarga de datos a través de dos ficheros en formato geojson

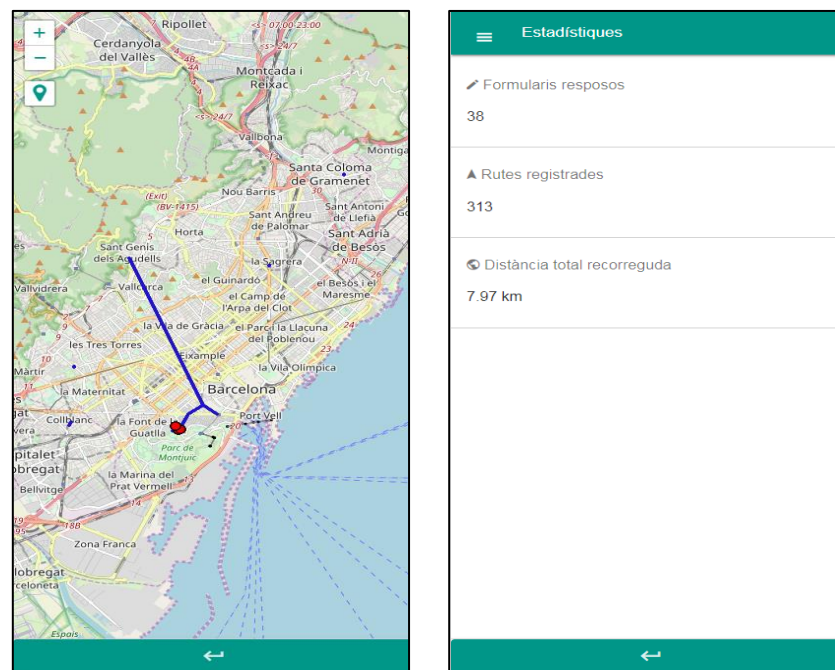


Figura 28 (izquierda). Vista del mapa de Barcelona con las rutas y formularios registrados por el usuario. Figura 29 (derecha). Estadísticas del usuario.